Ecole d'ingénieur de Genève HES-EIG
Département de l'unité d'enseignement
et de recherche UER4

Session automne 2003

Régulateur RST Appliqué au Réglage de Température

Candidat :

MAIRE Sébastien
Filière: Informatique

Professeurs responsables:

Jean-Marc ALLENBACH
Denis BICHSEL
En Collaboration avec:
CERN Mr Didier BLANC

Session de diplôme

e i g
Ecole d'ingénieurs
de Genève

Automne 2003

DIPLOME EN SYSTEMES ASSERVIS

REGULATEUR RST APPLIQUE AU REGLAGE DE TEMPERATURE

Descriptif :

Le CERN utilise, dans les accélérateurs de particules, des systèmes magnétiques qui doivent être maintenus dans des limites très étroites de température. Il s'agit de régler l'installation de refroidissement par circuit d'eau. Celle-ci est soumise à de fréquentes perturbations dues essentiellement à l'enclenchement et au déclenchement d'autres appareils qui y sont raccordés. On désire appliquer le régulateur le mieux adapté à garantir prioritairement une bonne immunité aux perturbations en gardant un bonne dynamique face aux variation de consigne. Pour cela, il faut bien connaître le processus. Comme celui-ci a des caractéristiques variables au cours du temps, on veut pouvoir mettre à jour à intervalles réguliers le modèle qu'on en a, sans cesser d'exploiter l'installation. On aura donc recours à un signal de perturbation approprié, dont les effets sur les grandeurs mesurables du système permettront d'ajuster les paramètres du modèle, qui permettront de calculer les coefficients du régulateur.

Fonctionnalités requises:

Bloc fonctionnel d'évaluation des paramètres du processus à régler.
Unité de dimensionnement des paramètres d'un régulateur RST.
Bloc fonctionnel de régulation RST.

Travail demandé :

- Etude du procédé d'identification de processus par perturbation calibrée superposée au signal de commande, et choix de cette perturbation adaptée au processus de refroidissement.
- Programmation du bloc d'identification et implantation sur l'automate.
- Elaboration du modèle avec le programme. Mesures comparatives du modèle et de l'installation réelle à fin de validation.
- Application de dimensionnement de régulateur RST sur la base du modèle développé.
- Vérification du comportement dynamique du système global en simulation.
- Programmation du bloc de régulation RST et implantation sur l'automate.
- Validation de régulation en vraie grandeur.
- Mémoire en 4 exemplaires.
- Résumé sur une page.
- Logiciel final sur CDROM avec mode d'emploi.

ÉCOLE D'INGENIEURS * RÉPUBLIQUE ET CANTON DE GENÈVE * DIRECTION

Unité d'enseignement
et de recherche UER4

Classe : IN3

Candidat :

M. MAIRE SEBASTIEN

Filière : Informatique

Professeurs responsables :

Jean-Marc Allenbach

Denis Bichsel

En collaboration avec :
CERN M. Didier Blanc

IN-Maire-Allenbach/mb

REMERCIEMENTS :

Je tiens à remercier en premier lieu M. Jean-Marc ALLENBACH pour avoir collaboré avec moi dans tout le déroulement du projet de diplôme. Vous m'avez aidé dans tous les points (informations, compléments de cours, stratégies à prévoir) mais surtout le soutien lors de la rencontre de problèmes.

Je tiens à remercier aussi M. Didier BLANC pour la proposition du projet de diplôme à effectuer au CERN. J'ai appris beaucoup de choses intéressantes au sujet de la climatisation et ventilation (circuit de refroidissement).

Je tiens à remercier enfin M. AUBERSON André, M. JENA Helmut, BRAHY Jacky et tous ceux de la section ST / CV.

TABLE DES MATIERES :

1. INTRODUCTION p1
1.1 Presentation du CERN p1
1.2 Descriptif du projet p3
1.3 Travail demandé p3

2. IDENTIFICATION DU PROCESSUS DE L'ECHANGEUR DE TEMPERATURE p4
2.1 Procédé de l'identification d'un processus p4
2.2 Etude du processus de l'échangeur de température p6
2.3 Cas de l'échangeur de température du BA4 p8
2.3.1 Etude du premier système p8
2.3.2 Etude du second système p13

3. DIMENSIONNEMENT DU REGULATEUR RST DE L'ECHANGEUR DE TEMPERATURE p15
3.1 Définition du régulateur RST p15
3.2 Robustesse en stabilité et robustesse en performance p18
3.3 Méthodes pour le dimensionnement du régulateur RST du processus p24
3.3.1 Première méthode de dimensionnement : LoopShap CERN p24
3.3.2 Deuxième méthode de dimensionnement : méthode Landau p29
3.4 Analyse des résultats obtenus des deux méthodes p39

4. IMPLANTATION DU SYSTEME REGLE SUR L'AUTOMATE S7 p41
4.1 Présentation du logiciel SIMATIC S7 V5.2 p41
4.2 Programmation sur SIMATIC S7 V5.2 du système réglé p44

5. CONCLUSION p45

ANNEXE N°1 p47

ANNEXE N°2 p48

ANNEXE N°3 p49

BIBLIOGRAPHIE p50

1. INTRODUCTION:

1.1 Presentation du CERN:

Le CERN est l'Organisation Européenne pour la Recherche Nucléaire, le plus grand centre de physique des particules du monde. Ici les physiciens viennent explorer ce dont la matière est faite et quelles forces assurent la cohésion.

Le CERN existe avant tout pour leur fournir les outils nécessaires. Ce sont des accélérateurs, qui accélèrent les particules jusqu'à une vitesse proche de celle de la lumière et des détecteurs pour rendre les particules visibles.
Fondé en 1954, le laboratoire a été l'une des premières entreprises commune à l'échelle européenne et il comprend maintenant 20 états membres.
Le CERN comprend 20 Etats membres Européens, mais de nombreux pays non européens participe également de diverses façons.

Les Pays membres ont des devoirs et privilèges spéciaux. Ils contribuent au capital et aux frais d'exploitation des programmes du CERN, et sont représentés au Conseil, responsable de toutes les décisions importantes concernant l'Organisation et ses activités.

Certains états (ou organisations internationales) pour lesquels l'adhésion n'est pas possible ou pas encore réalisable participent en tant qu'observateurs. Le statut d'observateur permet aux états non membres d'assister aux réunions du Conseil et de recevoir des documents du Conseil, sans prendre part aux procédures de décision de l'organisation.

Les états observateurs et les organisations participant actuellement aux programmes du CERN sont la Commission Européenne, Israël, Japon, Fédération de Russie, Turquie, UNESCO et les Etats-Unis.

Les états non membres participant actuellement aux programmes du CERN sont Algérie, Argentine, Arménie, Australie, Azerbaïdjan, Belarus, Brésil, Canada, Chine, Croatie, Chypre, Estonie, Géorgie, Islande, Inde, Iran, Irlande, Mexique, Maroc, Pakistan, Pérou, Roumanie, Serbie, Slovénie, Afrique du Sud, Corée du Sud, Taiwan et Ukraine.

Des scientifiques de 220 instituts et universités d'états non membres utilisent également les installations du CERN. Les physiciens et les universités ou instituts de recherche qui les financent sont responsables du financement, de la construction et de la mise en œuvre des expériences auxquelles ils collaborent. Le CERN investit une part importante de son budget dans la construction de nouvelles machines (telles que le grand collisionneur de hadrons LHC) et il peut contribuer partiellement au coût des expériences.
Le CERN est actuellement composé des divisions suivantes :

- Division AB : Accélérateur et Faisceaux
- Division AC : Directoire des Accélérateurs
- Division AS : Support Administratifs
- Division AT : Technologie des Accélérateurs
- Division DSU : Unité de Support au Directoire
- Division EP : Physique Expérimentale
- Division EST : Soutien en ingénierie et Technologie
- Division ETT : Education et Transfert de Technologie
- Division FI : Finances
- Division HR : Ressources Humaines
- Division IT : Technologie de l'Information
- Division SPL : Approvisionnement et Logistique
- Division ST : Support Technique
- Division TH : Physique Théorique
- Division TIS : Inspection Technique et Sécurité

Les divisions sont aussi composées de différents groupes. Plus particulièrement, celle qui est intéressante est la division ST.
Elle se compose des groupes suivants :

- AA/AC
- AA/AS
- CE
- CV
- EL
- HM
- MO
- TFM

Le groupe CV (Climatisation-Ventilation) joue un rôle important à la construction et maintenance des systèmes de refroidissement et de ventilation pour tous les accélérateurs et leurs zones d'expériences au CERN.
C'est dans la section ST / CV que le projet va se dérouler.

1.2 Descriptif du projet :

Le CERN dispose actuellement d'un grand réseau de refroidissement qui est dispersé sur différents points du LHC (The Large Hadron Collider Project). Lors des expériences effectuées sur le faisceau du LHC, les accélérateurs de particules ainsi que les aimants doivent être maintenus dans des limites très étroites de température. Cette régulation est assurée par les échangeurs de températures.
Il s'agit de modéliser un régulateur spécifique de température garantissant la bonne immunité face aux variations de perturbations et gardant une bonne dynamique face à la variation de consigne au cours du temps. Il est important, dans ce cas, de connaître précisément le processus. Comme les caractéristiques de ce processus sont variables au cours du temps, il est important de pouvoir mettre à jour, à intervalles réguliers, le modèle approprié sans cesser d'exploiter l'installation. Ce système aura recours à un signal de perturbation approprié dont les effets sur les grandeurs mesurables de celui-ci permettront de calculer les coefficients du régulateur.

Fonctionnalités requises :

Bloc fonctionnel d'évaluation des paramètres du processus à régler
Unité de dimensionnement des paramètres d'un régulateur RST
Bloc fonctionnel de régulation RST

1.3 Travail demandé:

Le déroulement du projet se compose en 7 étapes :

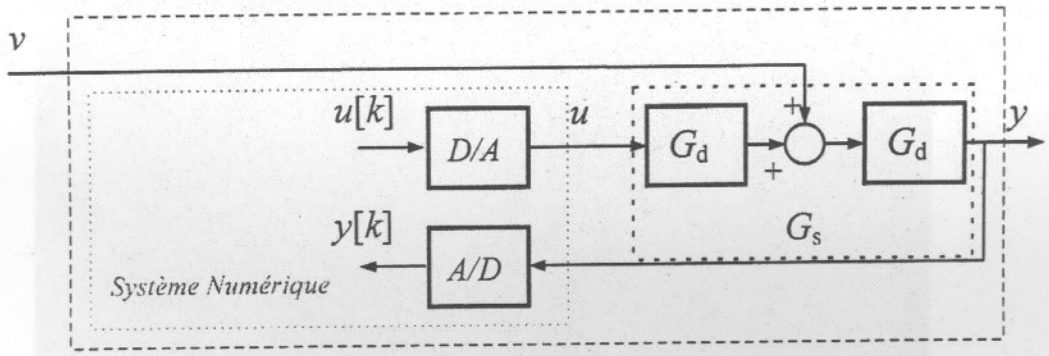
- Etude du procédé d'identification du processus de l'échangeur de température par perturbation calibrée au signal de commande, et choix de cette perturbation adaptée à ce processus.
- Programmation du bloc d'identification et implantation sur l'automate.
- Elaboration du modèle avec le programme. Mesures comparatives du modèle et de l'installation réelle à la fin de validation.
- Application de dimensionnement de régulateur RST sur la base du modèle développé.
- Vérification du comportement dynamique du système global en simulation.
- Programmation du bloc de régulation RST et implantation sur l'automate.
- Validation de régulation en vraie grandeur (Il n'est pas possible dans le cas du CERN de tester en grandeur réelle le régulateur RST car les expériences ne peuvent pas être interrompues)

2. IDENTIFICATION DU PROCESSUS DE L'ECHANGEUR DE TEMPERATURE :

2.1 Procédé de l'identification d'un processus :

Par définition, on identifie un processus S en supposant que l'on dispose d'un système d'acquisition de mesures qui peut agir sur le processus. Le système d'acquisition est le plus souvent informatisé, ce qui implique l'usage du calcul échantillonné et une acquisition à intervalles réguliers T. Les méthodes d'identification sont utilisées afin de donner une représentation mathématique d'un système réel lorsque l'analyse interne du processus n'est pas possible ou qu'elle ne donne pas de résultats exploitables. On peut procéder à deux identifications : en boucle ouverte ou en boucle fermée.

L'identification en ''boucle ouverte'' peut être représentée par la figure suivante.



Dans ce cas, l'idéal serait de pratiquer une mesure impulsionnelle du système. Il suffirait d'injecter à l'entrée u une impulsion de Dirac, puis de calculer la fonction de transfert par ''transformée inverse' du signal de sortie y.

$$Y(s) = G_s(s)\,U(s) \quad ou \quad Y(z) = G_s(z)\,U(z)$$

Si on implique en u(t) une impulsion de Dirac δ(t), la sortie y(t) est appelée g(t).

$$G_s(s) \;\bullet\!-\!\circ\; g(t) = g(kT) = g[k] \;\square\!-\!\blacksquare\; G_s(z)$$

Avec une mesure n finie d'échantillon de u[k] et y[k] ($0 \le k \le n$), on peut traduire la formule précédente dans l'espace temps par la convolution.

$$y[k] = g[k] * u[k]$$

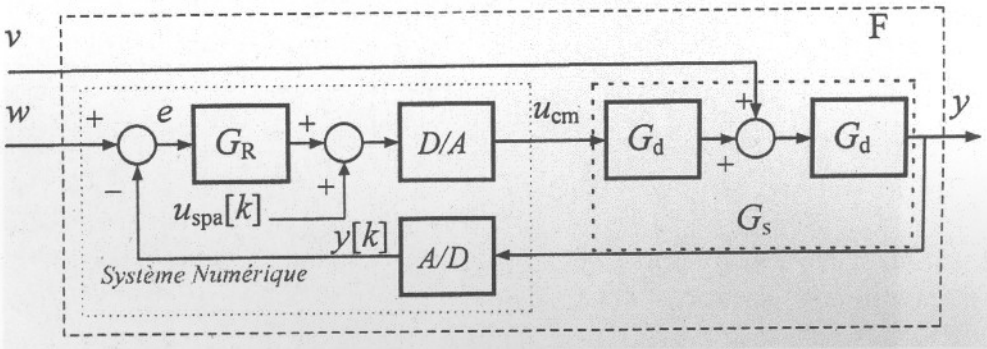
Après calcul, on peut obtenir g[k] par déconvolution, les n valeurs d'échantillons étant connues, et rechercher sa transformée en z pour obtenir la fonction de transfert.

$$y[k] */* u[k] = g[k] \quad \square - \blacksquare \quad G_s(z)$$

Cette méthode permet, certes, d'extraire la réponse impulsionnelle g[k] avec une assez bonne approximation mais il faut relever certains problèmes.

- Le signal y[k] peut subir les effets d'une perturbation υ (t), ce qui fausse la réponse impulsionnelle identifiée.
- On ne peut pas toujours pratiquer un essai sur une installation en ''boucle ouverte''.
- Le résultat final dépend du nombre d'échantillons.
- Le signal u(t) doit avoir une puissance suffisante pour mouvoir l'installation.

L'identification en ''boucle fermée'' peut être représentée par la figure suivante



On peut pratiquer une identification du système en cours de fonctionnement sans l'affecter. C'est pour cela que l'on superpose, au signal de sortie du régulateur, une séquence binaire pseudo aléatoire uspa[k] de valeur moyenne nulle et de faible amplitude. Dans ce cas, on identifie le système en boucle fermée Gaf.

$$G_{af} = \frac{G_s}{1 + G_R\, G_s}$$

Après identification, il est possible de reconstituer le système à régler car on connaît le régulateur utilisé.

$$G_s = \frac{G_{af}}{1 - G_R\, G_{af}}$$

A partir du System d'Identification Toolbox de Matlab, on peut déterminer à l'aide des mesures réelles des entrées et sorties du système la fonction de transfert du processus.

$$G_{af}(z) = \frac{b_0 + b_1 z^1 + \cdots + b_{n-1} z^{n-1} + b_n z^n}{1 + a_1 z^1 + \cdots a_{n-1} z^{n-1} + a_n z^n}$$

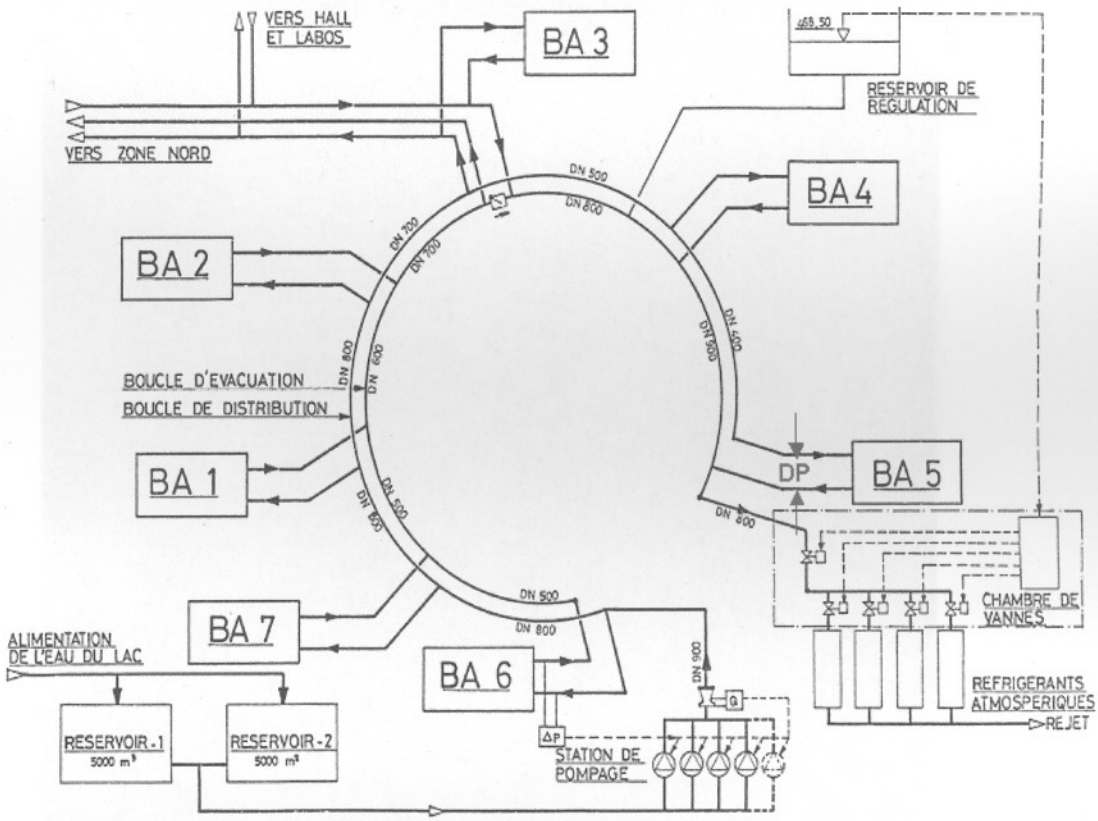
On peut obtenir, grâce à cette méthode d'optimisation, la relation entre l'entrée et la sortie du système.

$$Y(z) = G_{af}(z)\,U(z) = \frac{b_0 + b_1 z^1 + \cdots + b_{n-1} z^{n-1} + b_n z^n}{1 + a_1 z^1 + \cdots a_{n-1} z^{n-1} + a_n z^n} U(z)$$

A ce moment, le processus a été identifié.

2.2 Etude du processus de l'échangeur de température :

Afin d'étudier le processus d'identification, il est important de connaître le fonctionnement de l'échangeur de température.
Dans le cadre de la restructuration des réseaux de distribution de l'eau à l'intérieur du CERN, l'accélérateur du SPS (Super Proton Synchrotron), et plus précisément les bâtiments auxiliaires BA's (voir ci-dessous la disposition des BA), doivent subir d'importantes modifications de fonctionnalité.



L'eau nécessaire au refroidissement des différents aimants de l'accélérateur SPS circule en boucle fermée. Son refroidissement s'effectue par les tours du BA6. Les circuits du SPS sont composés d'une régulation de température.
Sur chaque circuit primaire des bâtiments auxiliaires est installé une vanne de régulation. Cette vanne permet, en fonction de la température des circuits, d'alimenter les échangeurs de refroidissement des circuits secondaires.

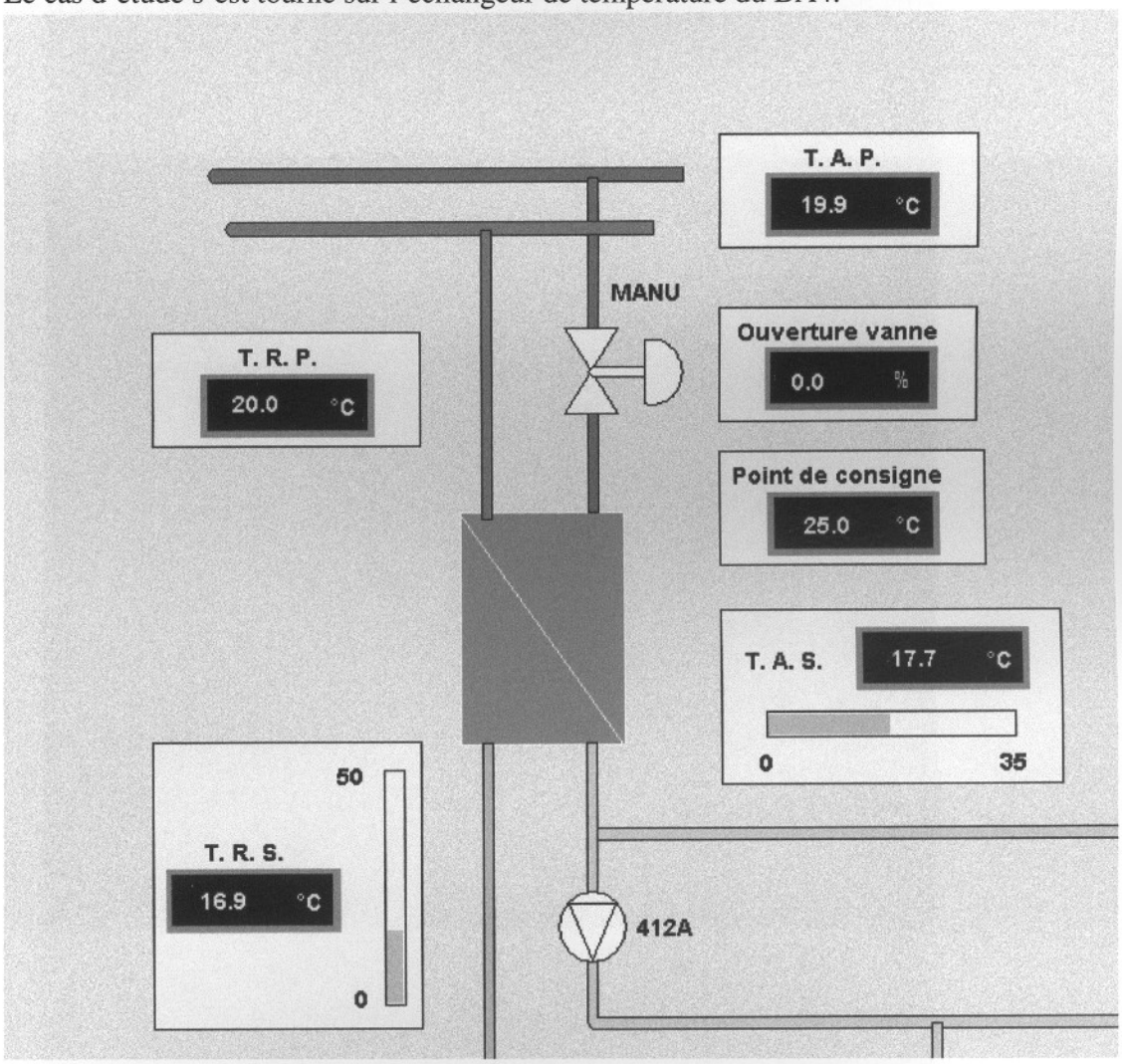
Cette chaîne de régulation de température se compose du matériel suivant :

- Sonde PT100
- Convertisseur ohm/milliampère
- Régulateur PID intégré à l'automate et géré par celui-ci
- Entrée et sortie analogique
- Vanne de régulation avec convertisseur électropneumatique

Le fonctionnement est le suivant :

- Point de consigne
- Valeur mesurée par la sonde
- Action par l'intermédiaire du régulateur sur la sortie analogique sur la vanne de régulation
- Régulation PID

Le cas d'étude s'est tourné sur l'échangeur de température du BA4.



2.3 Cas de l'échangeur de température du BA4 :

2.3.1 Etude du premier système :

Comme décrit dans le paragraphe précédent, la première approche de l'étude du processus de l'échangeur de température s'est orientée sur le rapport de mesure entre la vanne de régulation du circuit primaire (pourcentage d'ouverture de vanne) et la température de sortie sur le circuit secondaire (Température aller du circuit secondaire). Afin d'obtenir les mesures de ces deux variables, deux stratégies ont été utilisées.
Sur le réseau informatique du CERN, on peut disposer d'une application java ''JavaGuils'' qui permet de visualiser en temps réel les variations des paramètres de l'échangeur de température de chaque BA's. On a aussi la possibilité de consulter les archives des jours précédents stockées à la salle de contrôle TCR (dans l'enceinte principale du CERN) via cette application. Cependant, lors de la consultation des mesures, on a pu remarquer les périodes aléatoires de ces prises de mesures pour les différents paramètres. La raison vient de la configuration du système Wizcon qui permet d'enregistrer les moindres variations des paramètres d'entrée et de sortie de l'échangeur de température avec un écart d'erreur maximale de 0.5°C. En conséquence, si l'écart de variation est inférieur à cette limite, aucune donnée n'est enregistrée dans le but d'éviter la surcharge de mémoire dans le système Wizcon. Cette tentative a donc été abandonnée.
La deuxième stratégie a consisté à aller sur le terrain au BA4 et de modifier les paramètres de Wizcon pour une prise de mesure à intervalle de temps T fixe de 30s et un écart d'erreur de 0.1°C sur une durée de trois jours consécutifs.
A la réception des mesures du pourcentage d'ouverture de vanne et de la température via le réseau informatique du CERN, leurs analyses ont permis d'établir des représentations graphiques des deux paramètres en fonction du temps grâce au logiciel Microsoft Excel. Dans ce cas, il est important de définir un intervalle de mesure concluant pour l'identification du processus. En effet, il persiste souvent des coupures au sujet du système Wizcon ou du circuit de refroidissement du CERN.
Après avoir choisi la période de mesure mettant en bonne relation les paramètres d'entrée et de sortie (pourcentage d'ouverture de vanne et température aller du circuit secondaire), l'identification du processus grâce au System Identification Toolbox de Matlab peut commencer.

System Identification Toolbox est intégré dans le menu de Launch Pad de Matlab pour la version 6.0. Il est composé en deux parties. La première consiste à importer les données du processus à étudier. Il est plus aisé de réduire le nombre de valeurs de mesure (256 par exemple), de créer sur la ligne de commande Matlab les deux vecteurs correspondant à l'entrée u et la sortie y du processus étudié et de les convertir sous la forme donnée objet '' Data = iddata(y, u, Ts) '' qui correspond au format utilisé par System Identification Toolbox de Matlab. La variable y détermine le vecteur de mesure de sortie du processus, u détermine le

vecteur d'entrée du processus et la variable Ts détermine la période d'échantillonnage du processus. Dans notre cas d'étude, la variable u correspond au pourcentage d'ouverture de vanne, la variable y correspond à température aller du circuit secondaire et la période d'échantillonnage Ts est de 30 [s]. Après avoir importé les données dans l'environnement de l'identification de processus, à l'aide de l'option ''import'', une représentation graphique de l'entrée et de la sortie du processus est représentée en fonction du temps. L'option ''operation'' permet d'effectuer plusieurs fonctions dans l'objectif d'optimaliser l'étude du processus. Elle est composé de :

- ''Select channels'' : il permet de sélectionner les canaux si plusieurs entrées ou sorties ont été importées.
- ''Select experiments'' : il permet de sélectionner les différentes expériences importées.
- ''Merge experiments'' : il permet de fusionner plusieurs expériences importées.
- ''Select range'' : il permet, à partir des entrées et sorties du processus, de sélectionner une première partie du graphique pour l'analyse des données et une seconde partie pour la validation du modèle qui en résulte.
- ''Remove means'' : il permet de supprimer les valeurs moyennes des séquences des entrées et de sorties du processus.
- ''Remove trends'' : il permet d'estimer et de supprimer les tendances linéaires des signaux d'entrées et de sorties du processus. Il supprime les parasites relatifs au bruit et crée un signal plus fin (linéarité).
- ''Filter'' : il permet de filtrer les données d'entrées et de sortie du processus.
- ''Resample'' il permet de modifier la période d'échantillonnage sélectionnée lors de l'importations des mesures sur le plan de travail.

Dans le cas de l'étude du premier système, les fonctions utilisées ont été ''Remove means'', ''Remove trends''et ''Select range''. Après leurs exécutions, la partie importante de l'identification intervient. L'option ''estimate'', disponible sur System Identification Toolbox, permet d'estimer le modèle résultant de l'analyse des données d'entrées et de sortie du processus. Elle est composée de :

- ''Parametric models'' : il permet de générer des modèles dynamiques linéaire avec différentes structures, ordre du système et retard. La structure du modèle caractérise la relation entre les données d'entrées et de sorties du processus et entre les sources de bruit et les données de sortie. Les différentes structures du modèle qui sont à disposition sont : ARX, ARMAX, Output Error (OE), Box-Jenkins (BJ), State-Space.
- ''Spectral models'' : il permet d'estimer directement la réponse fréquentielle du système à partir des données importées sur l'espace de travail. Les méthodes utilisées sont : les techniques de la transformation de Fourier ou l'approche de Blackman-Tukey.

- ''Correlation models'': il permet d'estimer directement la réponse impulsionnelle du système en mettant en corrélation par filtrage les données d'entrées-sorties du processus.

Pour le premier système, la fonction utilisée a été ''Parametric models'' avec la structure State-Space et la recherche de la stabilité qui a permis de donner de meilleures résultats.
Le modèle State-Space peut être écrit de la façon suivante

$$x(t+1) = Ax(t) + Bu(t) + Ke(t)$$
$$y(t) = Cx(t) + Du(t) + e(t)$$

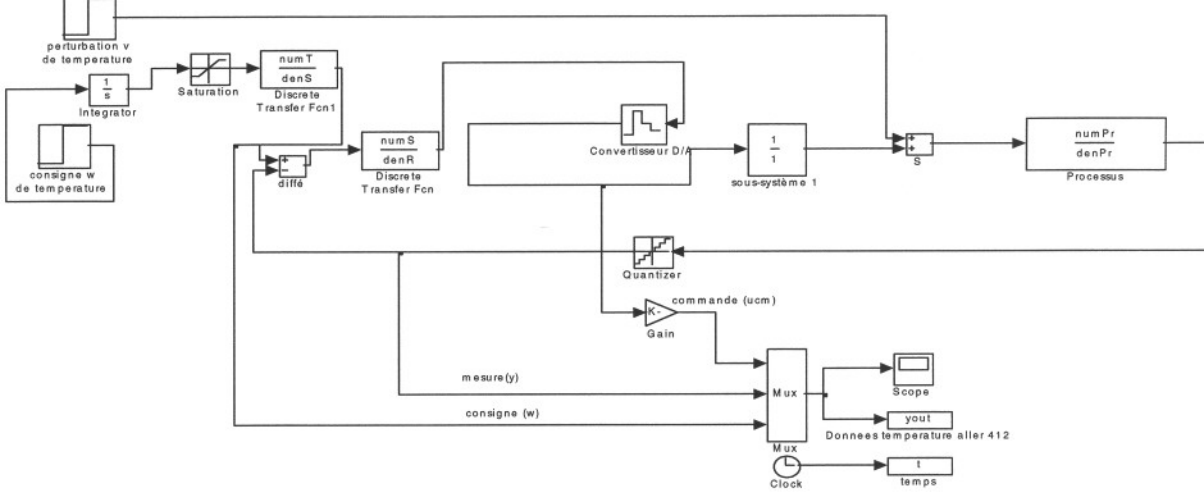
En effet, après avoir sélectionné cette fonction, on a la possibilité de choisir l'ordre du système afin d'obtenir la meilleure approximation du modèle. Elle est indiquée par la variable ''Best fits'' (mieux approprié en français) avec un pourcentage variant de valeurs négatives aux valeurs positives avec la limite de 100%. Plus on est proche de 100%, plus le modèle est proche de la réalité. En ce qui concerne le premier système, on a pu obtenir une approximation de l'ordre de 90%. On a la possibilité d'exporter la fonction de transfert du processus calculé, sous la forme State-Space, sur l'espace de travail de Matlab grâce à la fonction ''To workspace''. Un programme Matlab a été créé afin d'obtenir la fonction de transfert dans l'espace s (voir ci-dessous).

```
mc=d2c(pss2,'zoh');
A=mc.a; B=mc.b; C=mc.c; D=mc.d;
[nums,dens]=ss2tf(A,B,C,D);
sys1=tf(nums,dens);
```

pss2 correspond à la forme du modèle discret obtenue à partir de System Identification Toolbox. mc correspond à la conversion discret-continu de la variable pss2 avec la méthode ''zoh'' (zero-order hold) sur l'entrée. A la fin de l'exécution du programme, on a pu obtenir la fonction de transfert du processus (Pourcentage d'ouverture de vanne- température aller du circuit secondaire) suivante :

Transfer function:
```
  5.858e-006 s - 7.868e-008
----------------------------
s^2 + 0.0003156 s + 5.17e-006
```

Pour vérifier la validation du modèle, un modèle de simulation ''Regulateur_PID.mdl'' a été créé sur Simulink de Matlab.



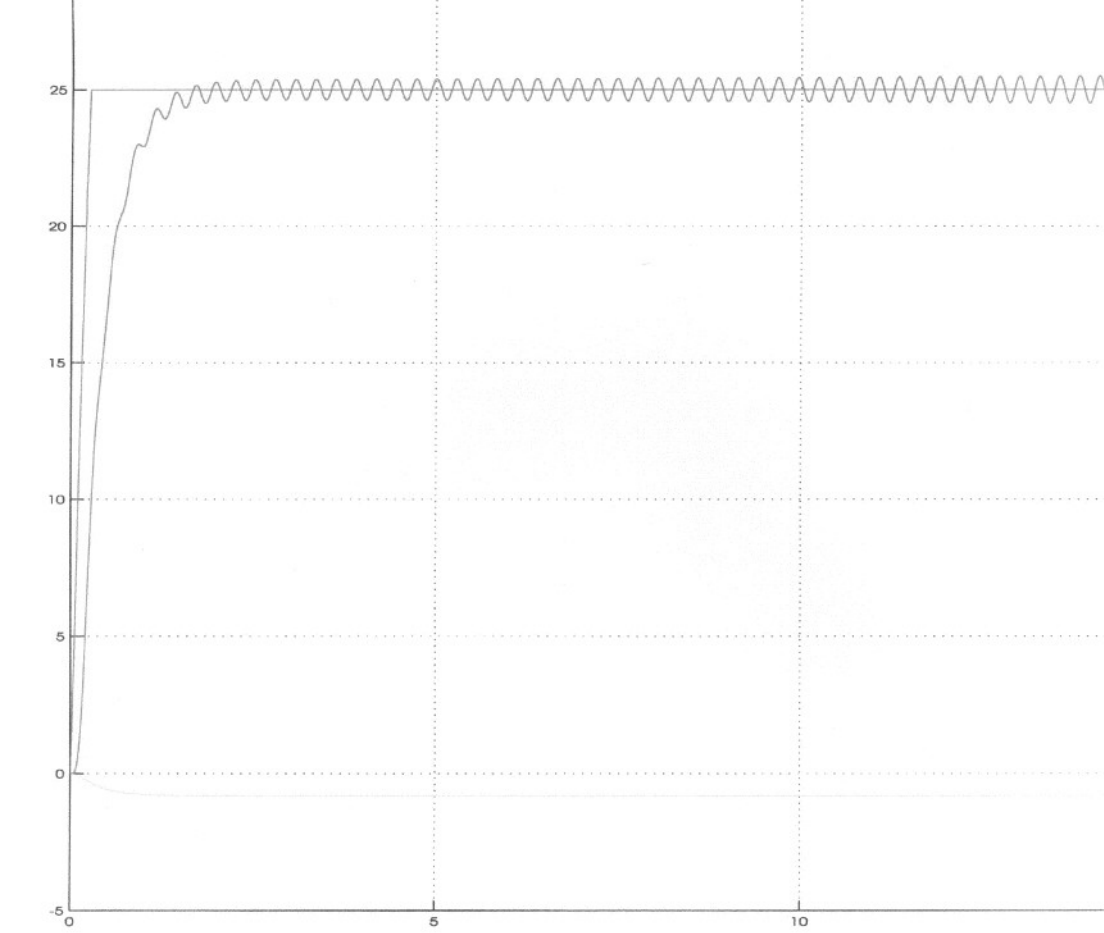
Les paramètres du régulateur PID digital, intégré actuellement dans les échangeurs de température du CERN, ont été utilisés pour la simulation (Gain proportionnel P = -2.0, Temps d'intégration I = 50000 [ms] et Temps de dérivation D = 5000 [ms]).

Pour information, la fonction de transfert du régulateur PID digital du CERN est :

Transfer function:

$$\frac{-2.77\,z^2 + 2.33\,z - 0.17}{z^2 - z}$$

On a pu remarquer que ce premier modèle a décrit un système instable (oscillatoire). La figure ci-dessous montre la réponse impulsionnelle de ce système.



La courbe bleue représente la consigne de l'échangeur de température à 25°C. La courbe en violet représente la réponse indicielle du régulateur PID digital du CERN.

L'étude de ce premier système a été une première tentative et ensuite abandonnée.
De plus, il a été communiqué par le CERN que les valeurs de pourcentage d'ouverture de vanne du circuit primaire n'étaient pas exploitables.

2.3.2 Etude du second système :

L'important dans l'étude de l'identification d'un processus est de connaître les relations propres entre chaque paramètre d'entrée et sortie. Afin d'optimiser l'identification du processus, une seconde variable d'entrée de l'échangeur de température a été utilisée : température aller du circuit primaire. En l'intégrant dans le système, on montre une relation plus fine et logique (température) entre les variables d'entrée et de sortie du processus. Les fichiers Excel ont été ainsi créés (voir annexe 1). Ils déterminent les variations de la température aller du circuit primaire et secondaire et du débit de la vanne du circuit primaire en fonction du temps pour une même période de prise de mesures.

Les méthodes pour le déroulement de l'identification de ce système sont identiques à l'étude du premier système. Cependant, l'importation des données d'entrée et de sortie a été modifiée. L'usage de la fonction Matlab ''iddata'' est différent (voir ci-dessous).

```
Data = iddata(y,[u1,u2],Ts);
```

La variable y signifie le vecteur de sortie du processus (température aller du circuit secondaire). Les variables u1 et u2 correspondent aux vecteurs d'entrée du processus (débit de la vanne et température aller du circuit primaire). La période d'échantillonnage Ts est restée inchangé et vaut 30 [s]. Ils correspondent aux valeurs les plus significatives de la journée lors de la prise des mesures.

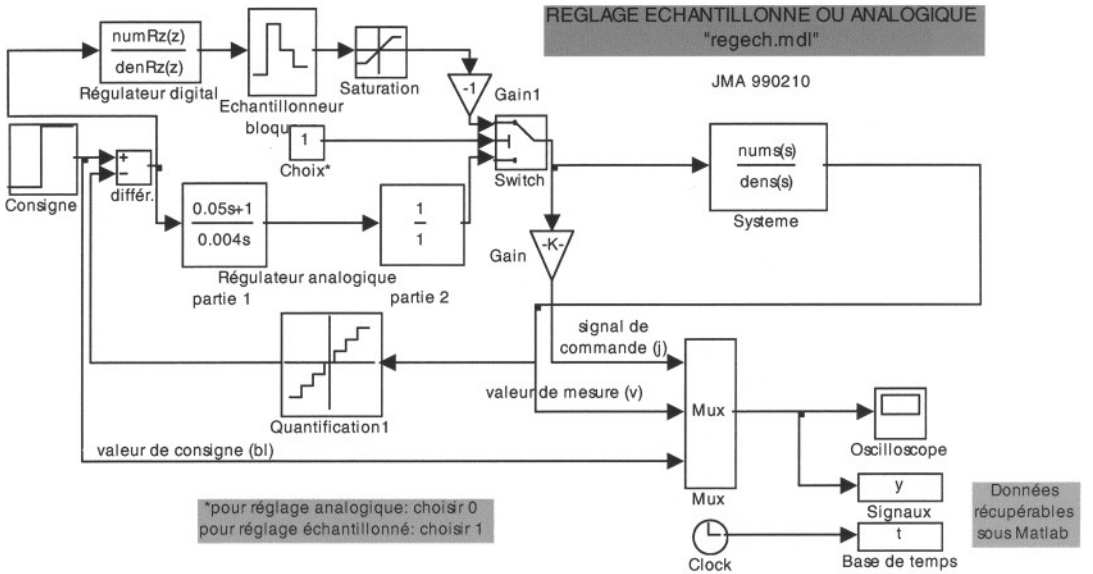
Après l'étude du modèle résultant, on a réussi à obtenir un ''Best fits'' de l'ordre de 80 % qui indique que le modèle est aussi très proche de la réalité. La méthode de conversion de ce modèle pour obtenir la fonction de transfert en continu de ce processus est presque identique au premier système (voir ci-dessous).

```
mc=d2c(pss3,'zoh');
A=mc.a; B=mc.b; C=mc.c; D=mc.d;
[nums,dens]=ss2tf(A,B,C,D,2);
sys2=tf(nums,dens) ;
[B,A]=cont2disc(nums,dens,Ts);
```

pss3 correspond à la forme du modèle discret du processus obtenue à l'aide de System Identification Toolbox. Après conversion dans l'espace s, on a pu obtenir la fonction de transfert du processus (débit de la vanne et température aller du circuit primaire – température aller du circuit secondaire) suivante :

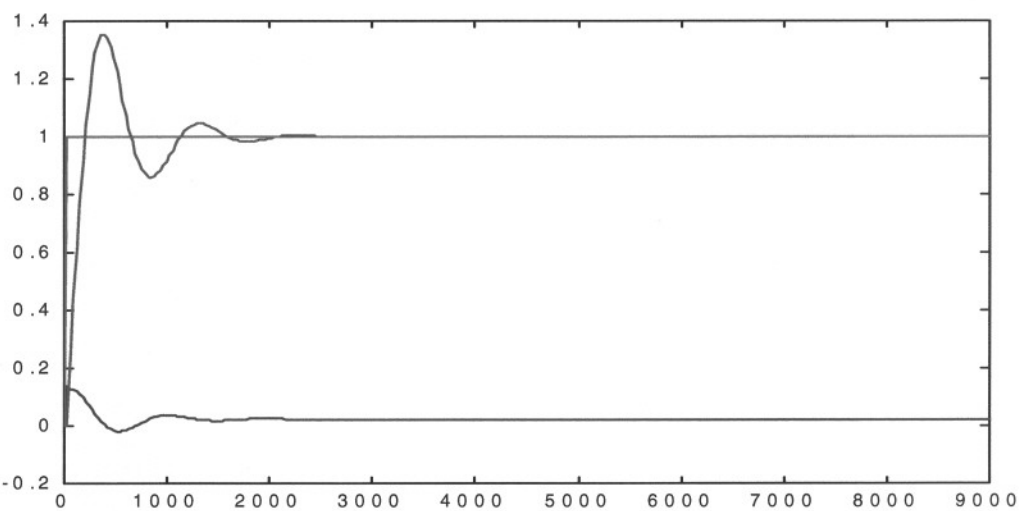
```
Transfer function:
  0.003669 s^2 + 7.797e-005 s + 3.209e-007
-------------------------------------------
s^3 + 0.03488 s^2 + 0.0001156 s + 1.481e-007
```

Pour vérifier la validité de ce modèle, le fichier Simulink ''regech_r.mdl'' a aussi été créé avec le même régulateur PID digital intégrer dans les échangeurs du CERN.



En appliquant cette fonction de transfert calculée pour le second système, les résultats ont été encourageants. Le système est stable et non oscillatoire par rapport au premier système.

La figure ci-dessous montre la réponse impulsionnelle du système avec le régulateur PID digital actuelle.

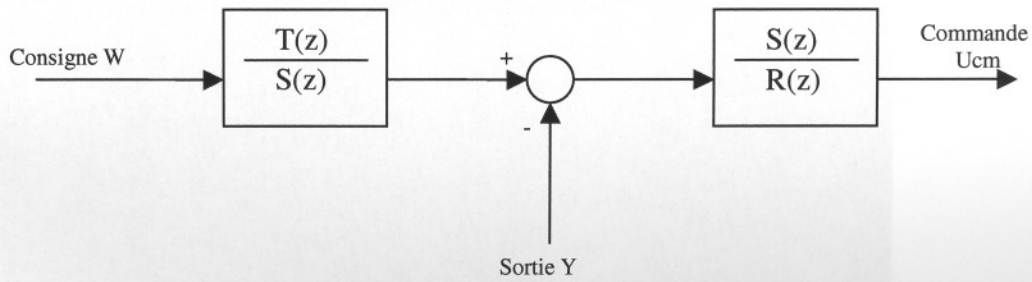


Ce processus, étant maintenant valide grâce à sa vérification avec le régulateur PID digital actuel, va permettre maintenant de déterminer les coefficients du régulateur RST. Le déroulement du projet se poursuit maintenant par le dimensionnement de ce régulateur.

3. DIMENSIONNEMENT DU REGULATEUR RST DE L'ECHANGEUR DE TEMPERATURE :

3.1 Définition du régulateur RST :

Le régulateur RST peut être semblable à un régulateur PID digital mais avec quelques avantages de plus. On utilise une structure qui agit différemment sur la consigne et sur la sortie. Pour obtenir un comportement différent pour une variation de consigne ou une perturbation, on peut appliquer un filtre polynomial sur la consigne.



Représentation des éléments composant le régulateur RST

Le régulateur RST est composé de deux parties :

- partie du régulateur interne à la boucle (PI,PID,RS) : $Gr(z) = S(z) / R(z)$
- partie externe pour garantir le comportement dynamique face à la consigne : $Glc(z) = T(z) / S(z)$

Une des racines de S(z) sert à compenser le zéro subsistant en boucle fermée, l'autre racine forme une paire avec celle de T(z) qui permet de diminuer ou d'augmenter le dépassement selon que la racine de T(z) est plus grande ou plus petite que celle de S(z). Dans tous les cas, le gain statique de Glc(z) doit être à 1.
On peut alors définir la loi de commande sous forme d'équation canonique suivante à partir du dessin précédent:

$$Ucm(z) = (T(z) / S(z)) * W(z) - (S(z) / R(z)) * Y(z)$$

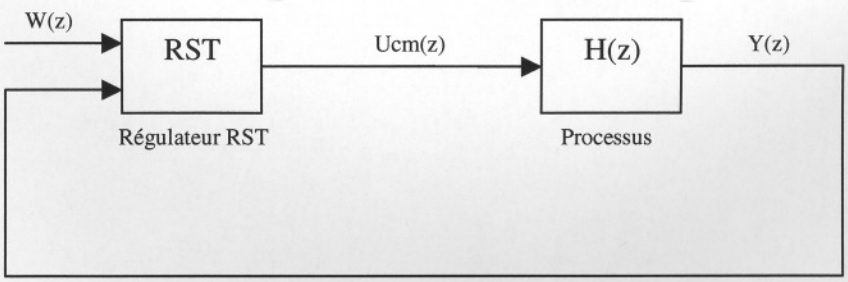


Schéma fonctionnel du modèle en asservissement avec un régulateur RST

H(z) est une fonction de transfert rationnelle strictement propre du processus. Elle est représentée par :

$H(z) = B(z) / A(z)$

Le degré du polynôme A(z) est strictement plus grand que celui du polynôme B(z). Dans la pratique, la fonction de transfert échantillonnée d'un processus à régler possède un surplus de pôles valant au moins 1, traduisant le fait que la réaction du système à une excitation quelconque s'observe au plus tôt après une période d'échantillonnage ; cette réalité physique provient des inerties et retards intrinsèques à tout élément dynamique.

On peut définir la fonction de transfert en boucle fermée par rapport à la sortie par la formule suivante :

$Y(z) = H(z) * ((T(z) / S(z)) * W(z) - (S(z) / R(z)) * Y(z))$

Les notions des polynômes R, S et T, introduites au début de ce chapitre, vont être maintenant développées. Selon la convention, les polynômes R, S, T se présentent sous la forme générale suivante :

$R(z) = r0 + r1 * z^{-1} + \ldots + ri * z^{(-i+1)}$ ou i désigne le degré du polynôme R

$S(z) = s0 + s1 * z^{-1} + \ldots sj * z^{(-j+1)}$ ou j désigne le degré du polynôme S

$T(z) = t0 + t1 * z^{-1} + \ldots tk * z^{(-k+1)}$ ou k désigne le degré du polynôme T
Il en est de même pour la fonction de transfert du processus appliqué H(z).
Les polynômes A et B sont définis par la forme suivante :

$A(z) = a0 + a1 * z^{-1} + \ldots + am * z^{(-m+1)}$ ou m désigne le degré du dénominateur A

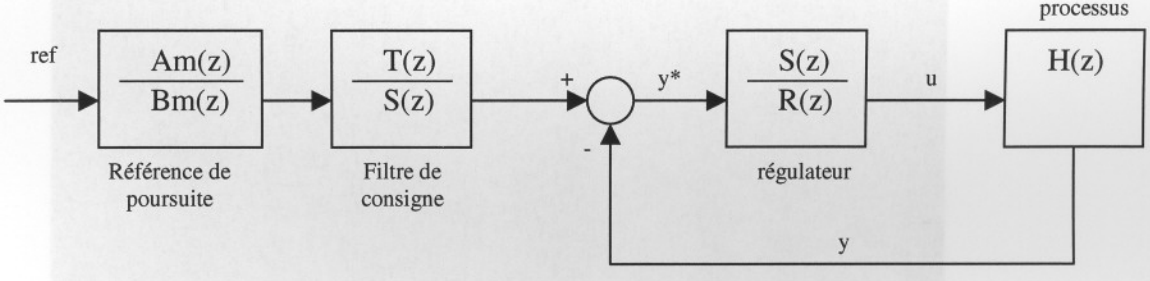
B(z) = b0 + b1 * z^-1 + … + bn * z^(-n+1) ou n désigne le degré du numérateur B

Afin de comprendre le fonctionnement du régulateur RST, il est évident de connaître son algorithme. On pourrait comparer le régulateur RST à un système de registres à décalage mais la suite va tout de suite éclairer le dilemme.

Par convention de notation, les variables utilisés vont être décrite sous la forme suivante :

- ref : vecteur contenant les valeurs du signal de consigne.
- ystar : vecteur contenant la sortie du modèle de référence de poursuite (y*).
- y : vecteur contenant les mesures de sortie du processus.
- u : vecteur contenant les organes de commande à l'entrée du processus.
- R,S,T : polynômes contenant les coefficients du régulateur.
- Am,Bm : polynômes du modèle de référence de poursuite.
- DegMax : dimension des tableaux utilisés pour stocker polynômes et variables.

La figure suivante montre les différents éléments qui interviennent dans l'algorithme de régulation RST.



Les entrées et sorties de l'algorithme de régulation RST sont définies par :

- En entrée : valeur de la mesure actuelle de la consigne : REFERENCE
  valeur actuelle de la sortie : YMESURE
- En sortie : commande calculée à appliqué au processus : COMMANDE

Ces paramètres sont appelés à chaque période d'échantillonnage.

Présentation de l'algorithme (langage de ADAPTECH):

```
| mise à jour des vecteurs
    for j = DegMax to 1 step -1
        | on progresse dans l'ordre chronologique
            y [j]   = y [j – 1]
            ref [j]  = ref [j – 1]
```

```
                    ystar [j] = ystar [j – 1]
                    u [j] = u [j – 1]
        next j

| prise en compte des valeurs en entrée
        y [0] =  YMESURE
        ref [0] = REFERENCE
| calcul de la nouvelle trajectoire de référence
        num = Bm [0] * ref [0]
        den  = 0
        | calcul de boucle
        for i = 1 to DegMax
                num = num + Bm [i] * ref [i]
                den = den + Am [i] * ystar [i]
        next i
        ystar = num – den

| calcul de la commande à appliquer
        sigmar = R [0] * y [0]
        sigmas = 0
        sigmat = T [0] * ystar [0]
        | boucle de calcul
        for i = 1 to DegMax
                sigmar = sigmar +  R [i] * y [i]
                sigmas = sigmas + S [i] * u [i]
                sigmat = sigmat + T [i] * ystar [i]
        next i
        u [0] = ( sigmat – sigmar – sigmas ) / S [0]
        COMMANDE = u [0]
| il ne reste plus qu'à appliquer cette commande
end.
```

Après avoir décrit le régulateur RST, il est important de connaître une nouvelle notion pour le dimensionnement de ce régulateur : la robustesse.

3.2 Robustesse en stabilité et robustesse en performance :

Le but de la synthèse robuste est le calcul d'un compensateur K(s) qui assure la stabilité su système en boucle fermée, et garantit un niveau de performance donné pour le système nominal seul, ou pour l'ensemble des systèmes perturbés. Cette distinction sur l'objectif de la compensation conduit à définir deux types de robustesse :

- on parle de système robuste en stabilité si ce système demeure stable en présence d'incertitudes telles que des erreurs de modélisation, les bruits de mesures ou les perturbations externes.
- un système est robuste en performance si les performances sont conservées en dépit des perturbations (les performances pouvant se mesurer en terme de temps de réponse du système, d'amortissement des modes flexibles, de découplage perturbations/sorties etc....).

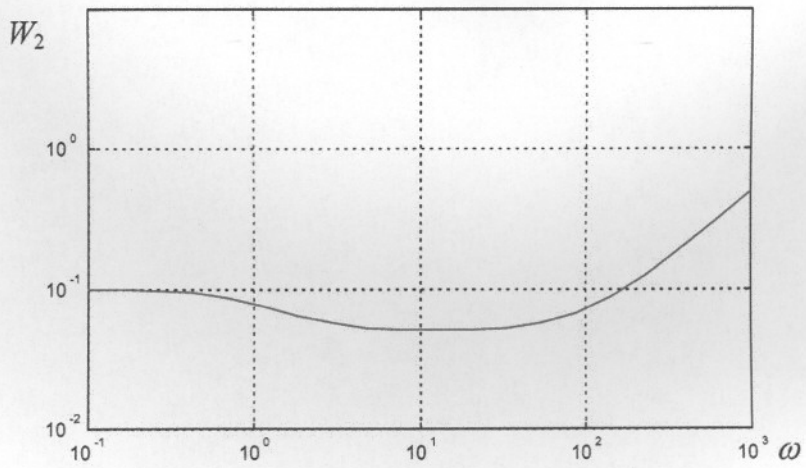
Ce dernier type de robustesse contient bien entendu le précédent, et l'on conçoit aisément que l'objectif de la synthèse robuste devra être d'assurer autant que faire se peut la robustesse en performance plutôt que la simple stabilité.

Il est important d'expliquer 3 notions importantes : Incertitude du modèle, Stabilité robuste et Performance.

Incertitude du modèle :

Dans les dimensionnements de régulateur, on a tenu compte des performances recherchées pour le système réglé et on a admis jusqu'ici que le système à régler était connu par sa fonction de transfert. On a souligné à multiples reprises que le système n'est connu qu'approximativement, mais sans en tenir compte explicitement dans les calculs. On se propose donc d'intégrer dans les calculs le fait qu'un système de fonction de transfert inconnue $G_s(s)$ est connu à peu près par la fonction de transfert modélisée $G_{s0}(s)$.
On peut exprimer un *profil d'incertitude* W_2 en fonction de la fréquence :

$$W_2(j\,\omega) = (G_s(j\,\omega) - G_{s0}(j\,\omega)) / G_{s0}(j\,\omega)$$



On peut interpréter le profil d'incertitude comme suit: La réponse harmonique $G_s(j\ \omega)$ se situe à l'intérieur d'un tube de rayon $|W_2\ G_{s0}(j\ \omega)|$ autour de la réponse harmonique $G_{s0}(j\ \omega)$ modélisée.

Stabilité robuste :

La définition de la stabilité dans le plan de Nyquist pour un système connu par sa réponse harmonique en boucle ouverte : *le système asservi est stable si on laisse le point "–1" sur la gauche en parcourant la réponse harmonique en boucle ouverte dans le sens des pulsations croissantes.*
On rappelle que la fonction de transfert en boucle ouverte est obtenue
par le produit de la fonction de transfert du régulateur avec celle du système à régler.

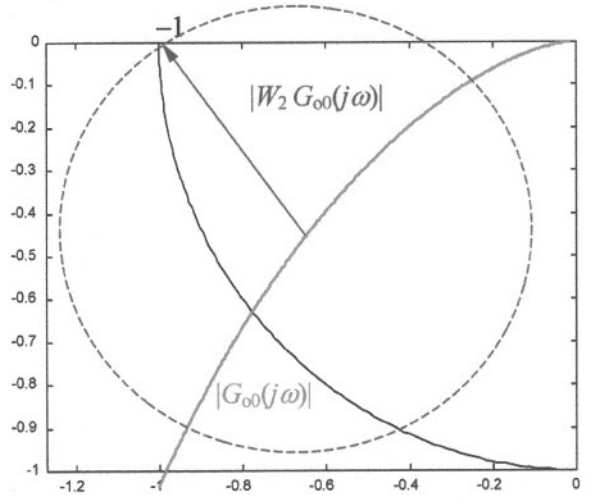
$G_o(s) = G_R(s)\ G_S(s)$

Il faut tenir compte de l'incertitude et définir la valeur "nominale" G_{o0} en boucle ouverte à partir du modèle G_{s0} du système.

$G_{o0}(s) = G_R(s)\ G_{S0}(s)$

On sait que les régulateurs dimensionnés d'après un modèle mathématique (fonction de transfert) ne produisent pas forcément un réglage robuste. On définit la *robustesse* comme l'insensibilité du comportement du système asservi face à des

variations des paramètres du processus à régler. Un système réglé est robuste si ses pôles et ses zéros bougent très peu lors d'une évolution du processus (usure, vieillissement, environnement thermique, ...). Pour tenir compte de l'incertitude, on doit adapter la définition en spécifiant que le point "–1" doit être laissé à gauche à une distance $|W2\ Go0(j\ \omega)|$ de la réponse harmonique nominale $Go0(j\ \omega)$.



La condition de *stabilité robuste* s'énonce en affirmant que la distance du point "–1" à la courbe de réponse harmonique nominale doit être supérieure au rayon d'incertitude, pour les pulsations de 0 à ∞.

$$|-1- Go0(j\ \omega)| = |1+ Go0(j\ \omega)| \bullet |W2\ Go0(j\ \omega)|$$

On peut traduire la condition de stabilité robuste pour la boucle fermée en divisant la relation précédente par la réponse harmonique nominale en boucle ouverte.

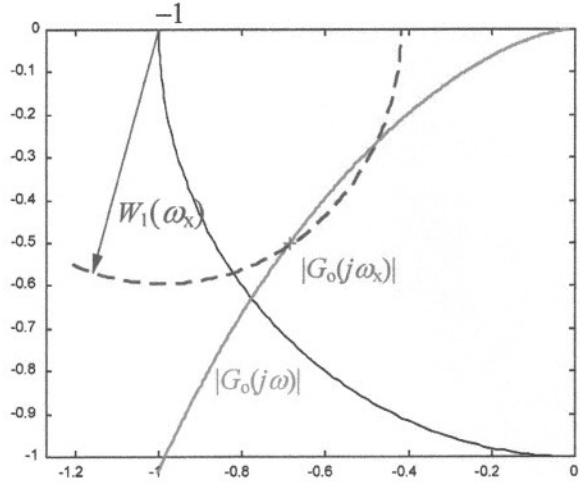
$$1 / |Gcf0(j\ \omega)| = |1+ Go0(j\ \omega)| / |\ Go0(j\ \omega)| \bullet |W2(j\ \omega)|$$

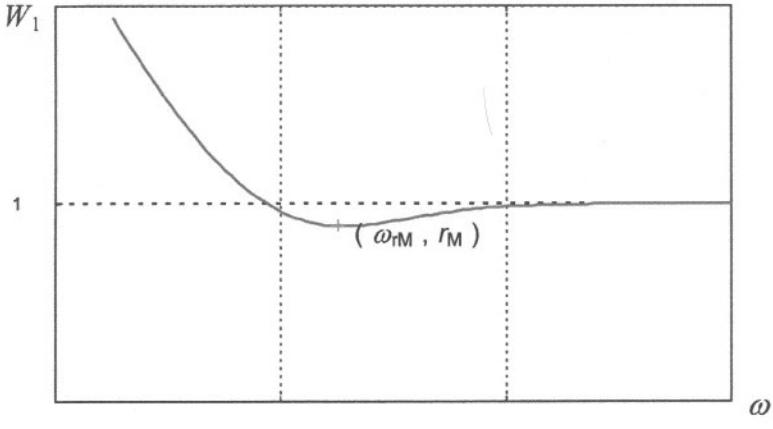
ou encore

$$1 \bullet |\ Gcf0(j\ \omega)W2(j\ \omega)|$$

Performance:

On définit la marge radiale r_M comme la plus courte distance entre la réponse harmonique et le point "–1". Il est aussi clair qu'on souhaite que cette distance soit très grande (idéalement infinie) lorsque la pulsation tend vers zéro. On se propose ici d'étendre ce concept à un rayon W_1 qui est une fonction de ω. Il est clair que le minimum de cette fonction est bien la marge radiale.



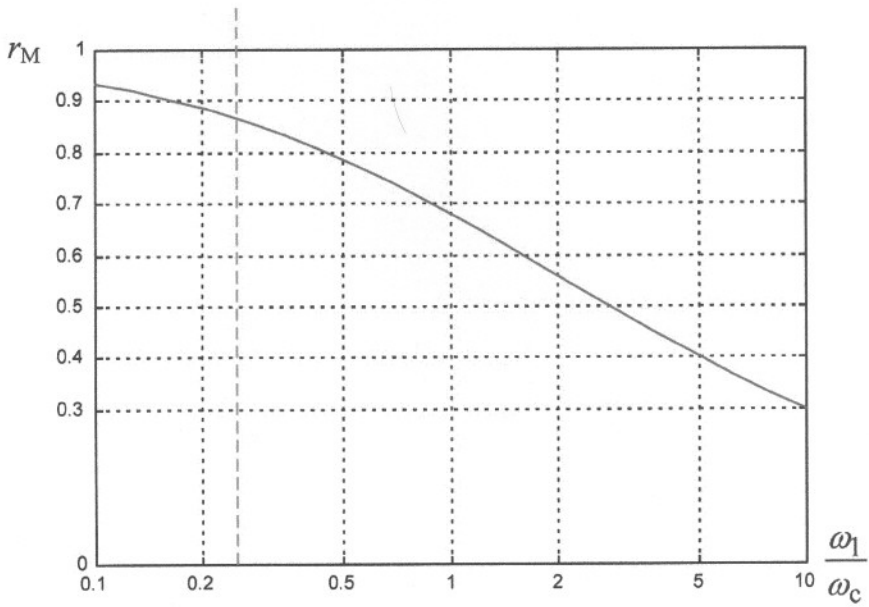


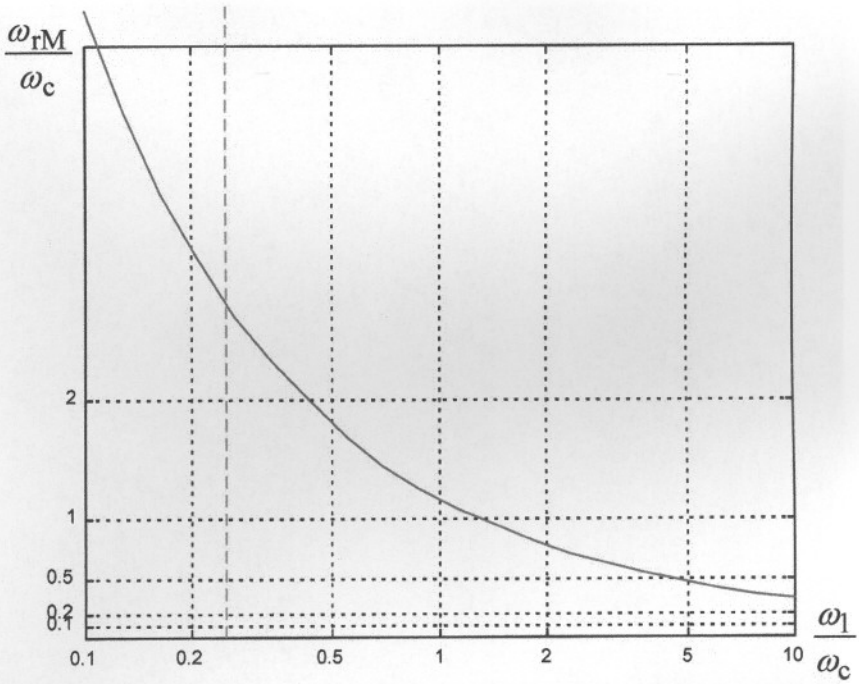
La condition de performance est que la distance du point "–1" à la réponse harmonique soit toujours supérieure à la fonction de performance W_1, pour toutes les pulsations :

$|1 + Go(j\omega)| \geq |W1(\omega)|$ pour tout ω

On retrouve bien les notions établies précédemment : une distance très élevée à faible pulsation pour garantir un écart statique très petit, une distance minimale à ne pas franchir et une distance qui tend vers 1 pour les pulsations élevées, lorsque la réponse harmonique en boucle ouverte tend vers 0. Idéalement, il faudrait une pente de –1 à basse pulsation pour garantir un écart statique nul, et même une pente de –2 pour les très
petites pulsations pour favoriser une rapide correction de l'effet d'une perturbation. La rapidité de convergence vers 1 aux pulsations élevées sera choisie en fonction du système et de son environnement : pulsations du système à conserver et pulsations de bruit à atténuer.

On a calculé la marge radiale et la pulsation correspondante ω_{rM} en fonction du rapport de pulsation ω_1/ω_c (voir abaque) pour établir le profil de performance en fonction du cahier des charges du système réglé.





La description des ces trois notions de robustesse permettent maintenant d'avoir une vision plus précise du réglage robuste. Dans le cadre du projet, deux méthodes de dimensionnement intégrant la robustesse ont été choisies.

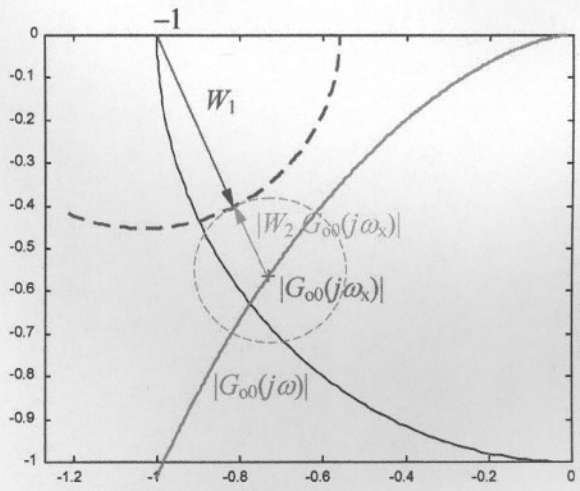
3.3 Méthodes pour le dimensionnement du régulateur RST du processus :

3.3.1 Première méthode de dimensionnement : LoopShap CERN

La notion de ''LoopShap CERN'' est une méthode se basant sur le façonnage de boucle (synthèse du réglage robuste).

Synthèse du façonnage de boucle :

Pour atteindre une performance robuste d'un réglage, il faut à la fois respecter simultanément la condition de stabilité robuste et le profil de performance. On peut interpréter cette double condition par l'exigence que pour toutes les valeurs de pulsation, le cercle d'incertitude et le cercle de performance soient disjoints, voire tangents.



Pour garantir une performance robuste, la distance entre le point "–1" et la réponse harmonique doit être inférieure à la somme des rayons de performance et d'incertitude.

$|1 + Go0(j\,\omega)| \geq |W1(\omega)| + |W2(\omega)\, Go0(j\,\omega)|$ pour tout ω

On doit donc construire une réponse harmonique $G_{o0}(j\,\omega)$ qui satisfait à la condition de stabilité robuste. Pour faciliter le calcul, on procède à l'approximation du module de la somme par le plus grand de ses deux termes.

$|Go0(j\,\omega)| \geq |W1(\omega)| + |W2(\omega)\, Go0(j\,\omega)|$ pour $1 < |Go0(j\,\omega)|$,

ω petit

$1 \geq |W1(\omega)| + |W2(\omega)\, Go0(j\,\omega)|$ pour $1 > |Go0(j\,\omega)|$,

ω grand

On peut récrire ces conditions en mettant la réponse harmonique $G_{o0}(j\,\omega)$ en évidence.

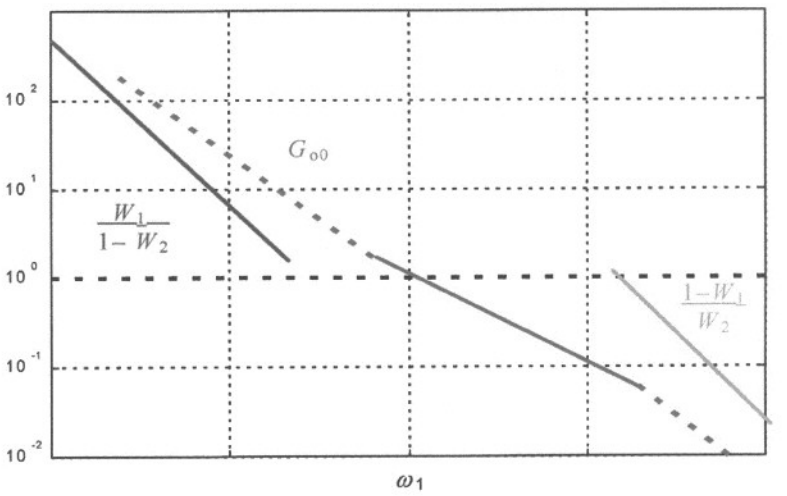
$|Go0(j\,\omega)| \geq |W1(\omega)| / (1 - |W2(\omega)|)$ pour $1 < |Go0(j\,\omega)|$,

ω petit

$|Go0(j\,\omega)| \bullet (1 - |W1(\omega)|) / |W2(\omega)|$ pour $1 > |Go0(j\,\omega)|$,

ω grand

On fera passer la réponse harmonique G_o en boucle ouverte entre les limites fixées, en s'arrangeant de franchir l'horizontal de module 1 avec une pente de –1. On procédera par adjonction successive de pôles et de zéros en ajustant à chaque fois le gain, jusqu'à ce que la réponse harmonique s'insère à notre satisfaction. On appelle cette étape le *façonnage de boucle* ouverte (en anglais *loop shaping*).



Il ne reste plus qu'à extraire la fonction de transfert du ulateur.

$G_R(s) = Go0(s) \;/\; Gs0(s)$

Comme le suggère l'équation ci dessus, le régulateur compense les pôles et zéros du système à régler et contient ceux qu'on a imposés lors du façonnage de boucle. Avant la réalisation, on peut être appelé à faire un compromis: si le système à régler compte des racines complexes conjuguées fortement oscillatoire, on corrigera celles du régulateur en adoptant la même pulsation propre, mais un meilleur facteur d'amortissement. Il y a lieu aussi d'être vigilant lorsque le retour n'est pas unité ou lorsque la grandeur réglée n'est pas exactement celle qu'on mesure.

Méthode utilisée dans le projet :

On effectue maintenant l'application pratique pour le projet.
On rappelle la fonction de transfert du système à régler modélisé de l'échangeur de température du BA4 sous la forme suivante :

Transfer function:
0.003669 s^2 + 7.797e-005 s + 3.209e-007
---
s^3 + 0.03488 s^2 + 0.0001156 s + 1.481e-007

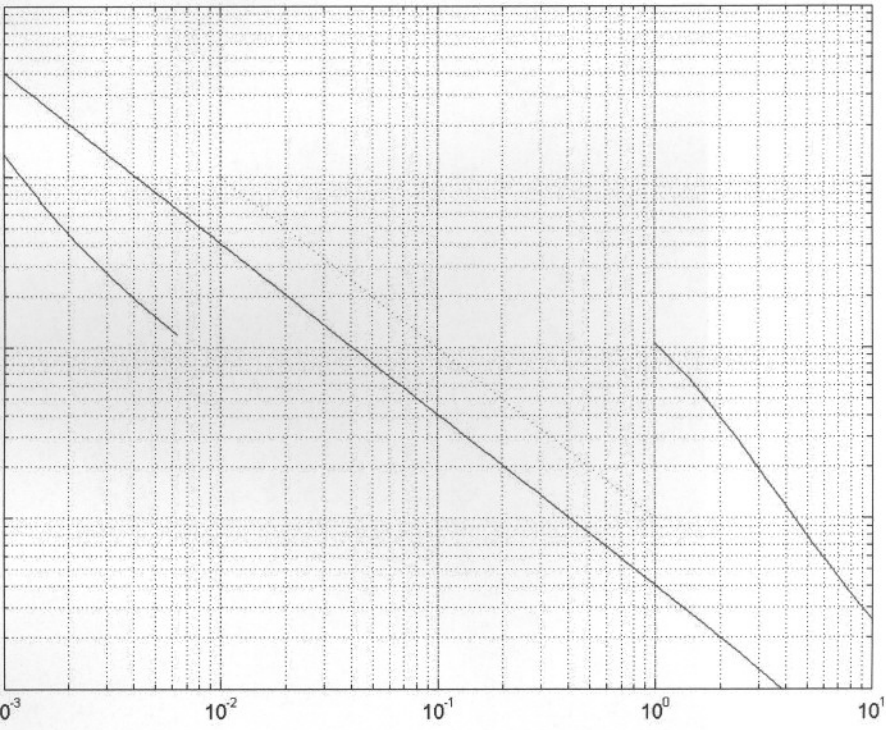
Le programme ''LoopShapCERN.m'' a été créé à cet effet (voir annexe 2 avec compléments de programme : ''paramCERN.m'', ''ouvdiscret.m'', ''cartediscret.m''). Ce fichier décrit la méthode du façonnage de boucle. Les étapes de calcul sont :

- Performance
- Incertitude
- Limite inférieure aux basses fréquences
- Limite supérieure aux hautes fréquences
- Système à régler
- Système en boucle fermée
- Calcul régulateur RST

Les résultats obtenus par cette méthode sont :

- Façonnage de boucle ouverte (en continu):

Il détermine un système à régler qui respecte le profil de performance et la condition de stabilité robuste. La figure ci-après le démontre :

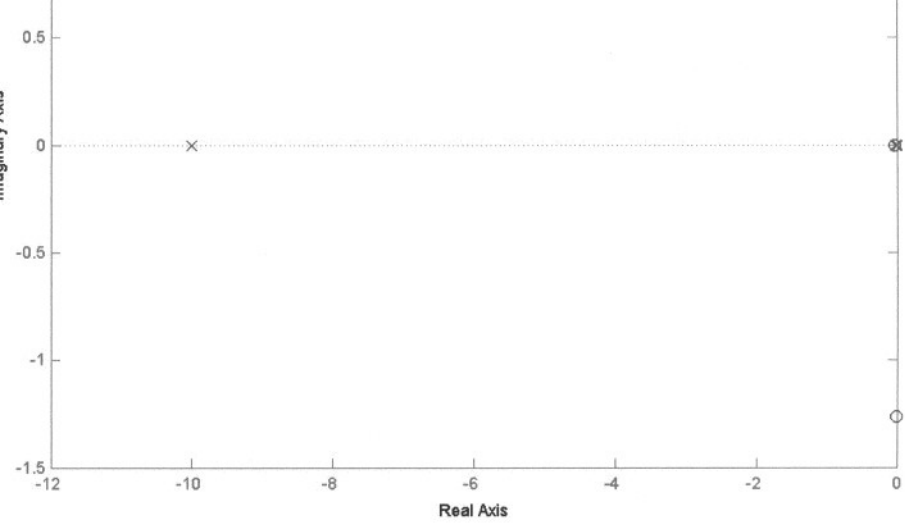


Fonction de transfert du régulateur RST LoopShap CERN

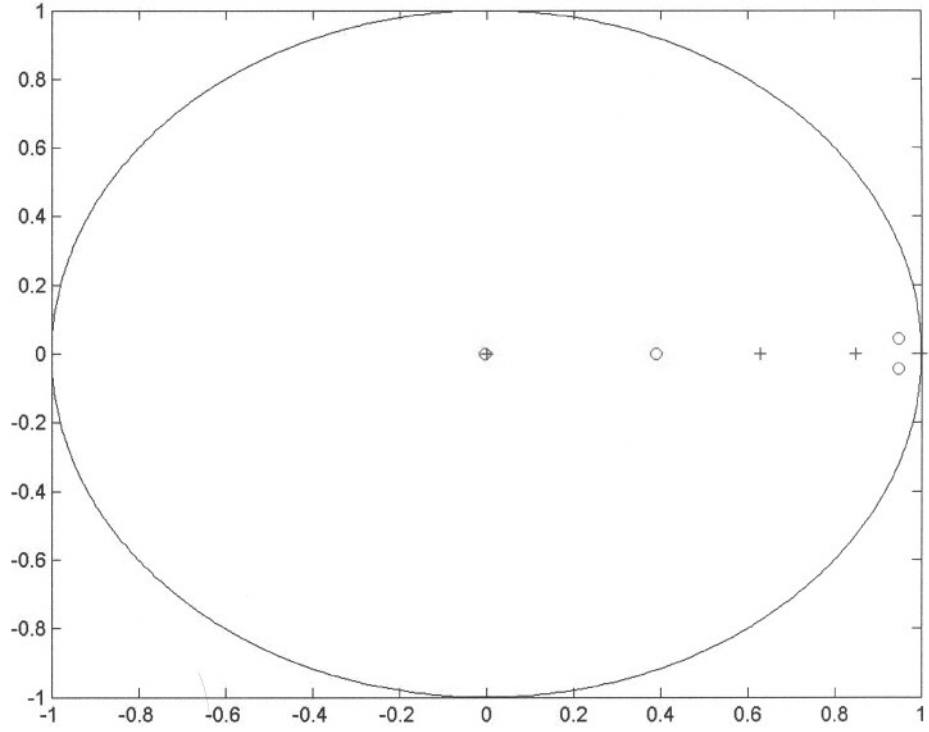
Transfer function:

$$\frac{0.4083\ z^3 + 0.01424\ z^2 + 4.721e\text{-}005\ z + 6.048e\text{-}008}{0.003669\ z^4 + 0.03677\ z^3 + 0.00078\ z^2 + 3.209e\text{-}006\ z}$$

Pôles et zéros régulateur RST LoopShap CERN (en
tinu:



Pôles et zéros régulateur RST LoopShap CERN (en discret):

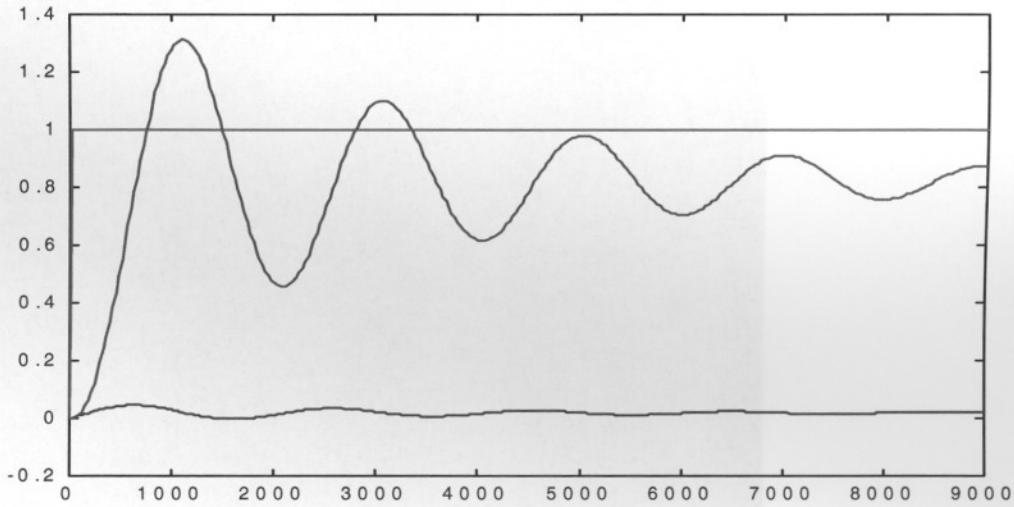


Il possède quatre zéros :

z1 = 9.4803 e-01 + 4.346 e-02
z2 = 9.4803 e-01 - 4.346 e-02
z3 = 3.8994 e-01
z4 = -3.3445 e-03
et trois pôles :

p1 = 1
p2 = 8.4736 e-01
p3 = 6.2996 e-01
p4 = 0

- Comportement dynamique avec régulateur RST LoopShap CERN :



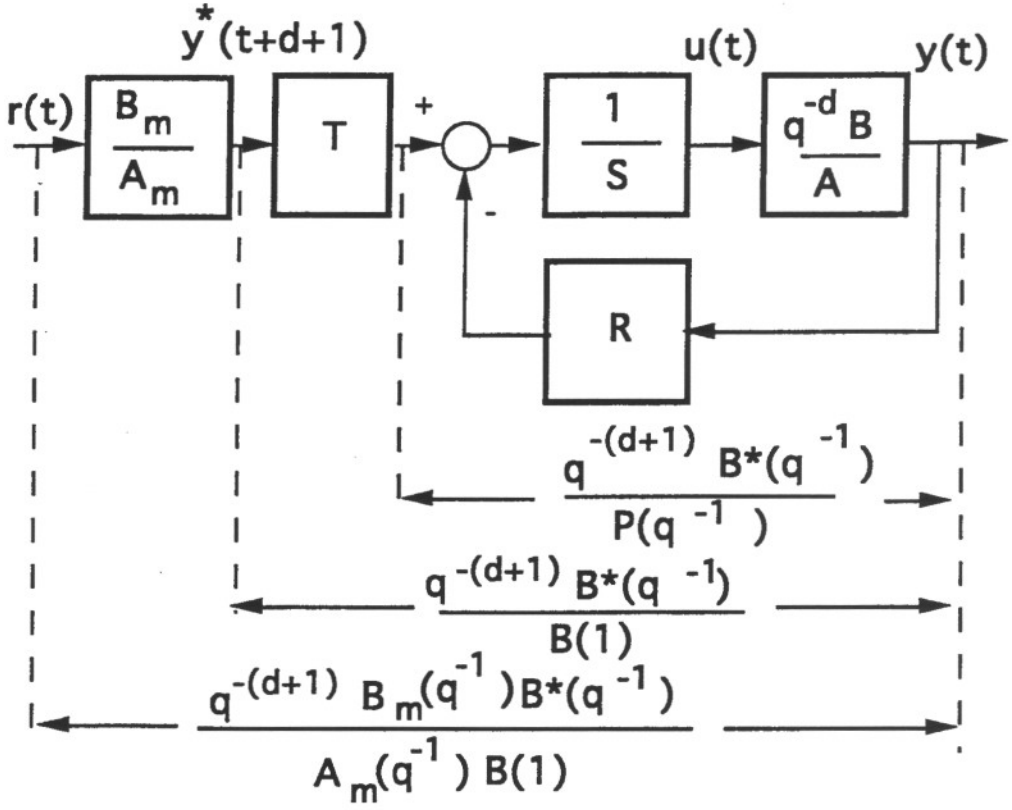
3.3.2 Deuxième méthode de dimensionnement : méthode Landau

La méthode Landau consiste à la technique du ''placement des pôles'' très répandue pour le calcul des régulateurs numériques destinés aux procédés mono entrée-mono sortie.
La procédure de calcul implique la spécification de pôles désirés de la boucle fermée (problème de ''stabilité nominale'') et de certaines parties fixes du régulateur pour le rejet exact des perturbations en régime stationnaire (intégrateur pour le cas des perturbations en échelon).

Néanmoins, les propriétés de rejection des perturbations à différentes fréquences (problème de ''performance nominale'') et la robustesse du système en boucle fermée vis-à-vis des incertitudes de modélisation (problème de ''stabilité robuste'') va dépendre des spécifications.
La fonction de sensibilité perturbation-sortie constitue un indicateur clé pour les performances nominales (et robustes) et la stabilité robuste du système.
Il est important d'expliquer maintenant la méthodologie du ''placement des pôles''.

Le placement de pôles :

La structure du système en boucle fermée échantillonnée est donnée dans la figure 1. Il est constitué par le procédé à commande échantillonné avec un bloqueur d'ordre zéro et le régulateur RST permettant de traiter indépendamment les problèmes de régulation de poursuite.



Le procédé échantillonné est décrit par l'opérateur de transfert :

$$H(q^{-1}) = \frac{q^{-d} B(q^{-1})}{A(q^{-1})} = \frac{q^{-d-1} B^*(q^{-1})}{A(q^{-1})}$$

Où q^-1 est l'opérateur de retard (y(t-1) = q^-1 y(t), d est le nombre entier de périodes d'échantillonnage contenues dans le retard pur du procédé et :

$$A(q^{-1}) = 1 + a_1 q^{-1} + \ldots a_{n_A} q^{-n_A}$$

$$B(q^{-1}) = b_1 q^{-1} + b_2 q^{-2} + b_{n_B} q^{-n_B} = q^{-1} B^*(q^{-1})$$

Les polynômes A(q^-1) et B(q^-1) sont supposés premiers entre eux. La forme canonique du régulateur RST est :

$$S(q^{-1})\, u(t) = T(q^{-1})\, y^*(t+d+1) - R(z^{-1})\, y(t)$$

Où u et y sont respectivement l'entrée et la sortie du procédé et y*(t+d+1) représente la trajectoire de référence désirée pour la poursuite.

La fonction de transfert en boucle fermée entre la trajectoire de référence et la sortie est donnée par :

$$H_{BF}(z^{-1}) = \frac{z^{-d} B(z^{-1}) T(z^{-1})}{A(z^{-1}) S(z^{-1}) + z^{-d} B(z^{-1}) R(z^{-1})}$$

$$= \frac{z^{-d} B(z^{-1}) T(z^{-1})}{P(z^{-1})}$$

où

$$P(z^{-1}) = A(z^{-1}) S(z^{-1}) + q^{-d} B(z^{-1}) R(z^{-1})$$

$$= P_D(z^{-1}) \cdot P_F(z^{-1}) = 1 + p_1 z^{-1} + p_2 z^{-2}$$

P(z^-1) définit les pôles en boucle fermée du système. PD(z^-1) correspond aux pôles dominants choisis en fonction des performances désirées du système en régulation et PF(z^-1) correspond aux pôles auxiliaires qui sont souvent introduits soit pour un effet de filtrage dans certaines zones de fréquence soit pour adoucir les variation de la commande, soit pour améliorer la robustesse du système en boucle fermée.

L'équation précédente a une solution unique (A(z^-1) et B(z^-1) étant premiers entre eux) pour :

$$nP = \deg P(z^{-1}) \leq n_A + n_B + d - 1$$
$$nS = \deg S(z^{-1}) \leq n_B + d - 1$$
$$nR = \deg R(z^{-1}) \leq n_A - 1$$

où

$$S(q^{-1}) = 1 + s_1 q^{-1} + \ldots s_{ns} q^{-ns} = 1 + q^{-1} S^*(q^{-1})$$

$$R(q^{-1}) = r_0 + r_1 q^{-1} + \ldots r_{nR} q^{-nR}$$

Pour des raisons variées (rejection de perturbation, blocage d'un signal, robustesse) les polynômes R(z^-1) et S(z^-1) contiennent en général des parties fixes spécifiées avant la résolution de l'équation.

C'est pour prendre en compte ces parties fixes pré-spécifiées que les polynômes R(z^-1) et S(z^-1) sont factorisés sous la forme :

$$R(z^{-1}) = R'(z^{-1})\ H_R(z^{-1})$$
$$S(z^{-1}) = S'(z^{-1})\ H_S(z^{-1})$$

Où HR(z^-1) et HS(z^-1) sont des polynômes pré-spécifiés.

L'équation en boucle fermée devient alors :

$$P(z^{-1}) = A(z^{-1})\, H_S(z^{-1})\ S'(z^{-1}) + z^{-d}\, B(z^{-1})\, H_R(z^{-1})\, R'(z^{-1})$$

La procédure de calcul se résume ainsi :

- Etape n°1 :
  Choix des pôles désirés en boucle fermée P(z^-1) = PD(z^-1) *PF(z^-1), des parties fixes du régulateur HR(z^-1), HS(z^-1).

- Etape n°2 :
  Résolution de l'équation des pôles en boucle fermée et obtention de R'(z^-1) et S'(z^-1). Détermination de R(z^-1) et S(z^-1).

- Etape n°3 :
  Calcul du pré compensateur T(z^-1) par T(z^-1) = G P(z^-1)

Où

G = 1/B(1) si $B(1) \neq 0$ ou G = 1 si B(1) = 0

Ceci conduit à la fonction de transfert en boucle fermée entre la trajectoire de référence et la sortie :

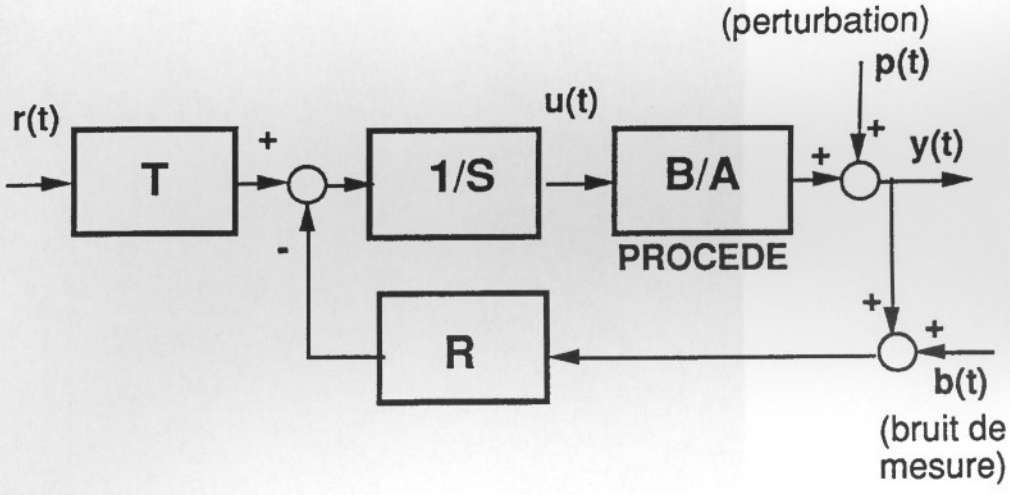
$$H_{BF}(z^{-1}) = z^{-d}\, B(z^{-1}) / B(1)$$

Pour la méthode du placement des pôles, on définit les fonctions de sensibilités.

Fonctions de sensibilité :

Le système en boucle fermée est sujet à des perturbations et il est important de connaître l'effet des ces perturbations sur la sortie et l'entrée du procédé. Par ailleurs, ces fonctions de sensibilité jouent un rôle fondamental dans l'analyse de la robustesse du système bouclé par rapport aux erreurs du modèle.
Deux types de perturbations seront considérées : les perturbations en sortie de procédé et les bruits de mesure.

A partir de la figure ci-dessous, on peut déterminer les différentes fonctions de sensibilité.



La fonction de transfert entre la perturbation p(t) et la sortie y(t) (fonction de sensibilité perturbation-sortie) est donnée par :

$$S_{yp}(z^{-1}) = \frac{A(z^{-1})\, S(z^{-1})}{A(z^{-1})\, S(z^{-1}) + z^{-d}\, B(z^{-1})\, R(z^{-1})} = \frac{A(z^{-1})\, S(z^{-1})}{P(z^{-1})}$$

La fonction de transfert entre la perturbation p(t) et l'entrée u(t) (fonction de sensibilité perturbation-entrée) est donnée par :

$$S_{up}(z^{-1}) = \frac{-A(z^{-1})R(z^{-1})}{A(z^{-1})S(z^{-1}) + z^{-d}B(z^{-1})R(z^{-1})} = \frac{-A(z^{-1})R(z^{-1})}{P(z^{-1})}$$

La fonction de transfert entre le bruit de mesure b(t) et la sortie du procédé y(t) (fonction de sensibilité bruit de mesure-sortie) est donnée par :

$$S_{yb}(z^{-1}) = \frac{-B(z^{-1})R(z^{-1})}{A(z^{-1})S(z^{-1}) + z^{-d}B(z^{-1})R(z^{-1})} = \frac{-B(z^{-1})R(z^{-1})}{P(z^{-1})}$$

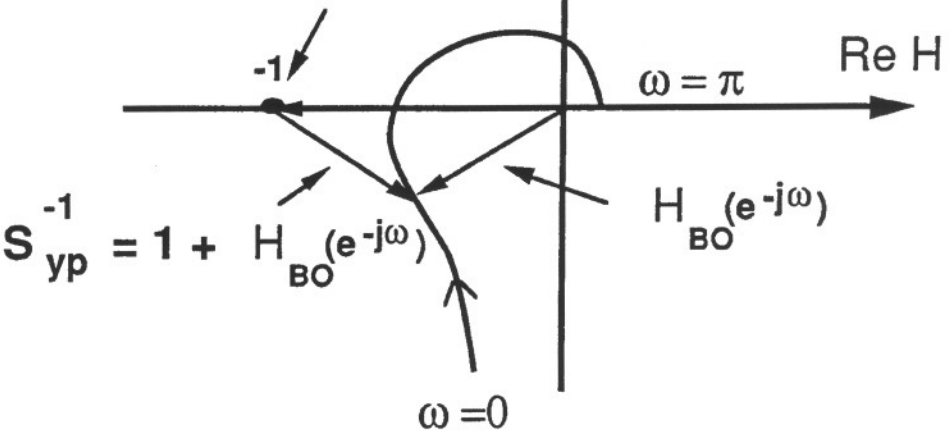
Notons une propriété importante :

$$S_{yp}(z^{-1}) - S_{ub}(z^{-1}) = S + T = 1$$

Il faut encore définir les marges de robustesse, incertitudes sur le modèle et stabilité robuste.

es marges de robustesse :

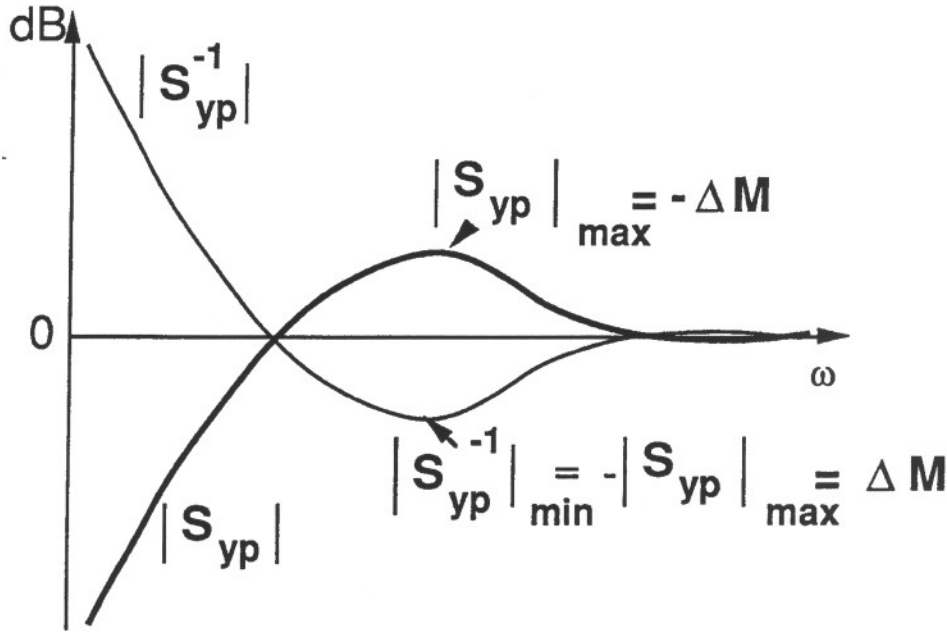
L'utilisation du lieu de Nyquist permet d'appréhender l'influence des erreurs de modèle et de déduire des spécifications pour la conception du régulateur RST afin d'obtenir une stabilité robuste pour certaines classes d'incertitudes su le modèle du procédé.



Le vecteur reliant l'origine à un point de l'hodographe de la fonction de transfert correspond à HBO(e^-jω) pour une certaine pulsation normalisée. Le point [-1,j0] sur le diagramme précédent est le point critique.

Rappelons que le critère de stabilité de Nyquist (cas où HBO(z^-1) est stable) exige que l'hodographe de la fonction de transfert en boucle ouverte HBO(z^-1) parcouru dans le sens des fréquences croissantes (de 0 ou 0.5fe) doit laisser à gauche le point critique.
Ce critère sera satisfait dans le cas du modèle ''nominal'' car les polynômes R et S seront calculés pour un P(z^-1) définissant les pôles de la boucle fermée qui est un polynôme ayant des racines asymptotiquement stables (P(z^-1) = 0 => |z| < 1).
L'éloignement minimal de l'hodographe de HBO(z^-1) par rapport au point critique va définir une marge de stabilité qui va dépendre du maximum de la fonction de sensibilité perturbation-sortie.

Les éléments suivant servent à caractériser l'éloignement de l'hodographe de HBO(z^-1) par rapport au point critique :
- Marge de module(ΔM) : Elle définit la valeur maximale admissible du module de la fonction de sensibilité perturbation-sortie.

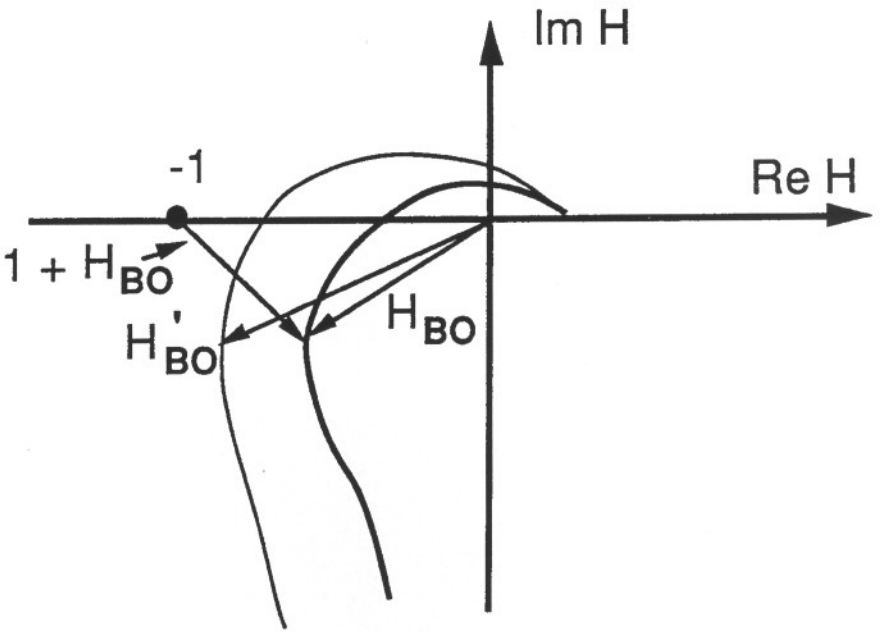


Marge de retard : Pour une certaine pulsation ω, le déphasage introduit par un retard pur τ est :

$\Phi(\omega) = \omega\tau$

On peut convertir la marge de phase $\Delta\Phi$ en marge de retard. C'est-à-dire calculer le retard additionnel qui ne conduira pas à l'instabilité. Il en résulte : $\Delta\tau = \Delta\Phi / \omega$

En ce qui concerne, l'incertitude sur le modèle et la stabilité robuste, on peut illustrer les effets des incertitudes ou des variations du modèle nominal sur le lieu de Nyquist en boucle ouverte par la figure suivante :



En règle générale, le lieu de Nyquist pour le modèle nominal se trouve à l'intérieur d'un ''tube'' correspondant aux tolérance possibles (ou acceptables) par rapport au modèle nominal.

Après avoir expliqué la théorie du placement des pôles, on peut l'illustrer dans le cas pratique (dans le projet).

Méthode utilisée dans le projet :

Pour évaluer le régulateur RST, on a utilisé l'application ''ppmaster.zip'' de Monsieur I.D. LANDAU disponible sur le site internet : http://www-lag.ensieg.inpg.fr/landau/bookIC/index_FR_TableDeMatieres_Annexes.htm. Cette application repose sur la théorie exposée précédemment (Je vous laisse la possibilité de la télécharger).

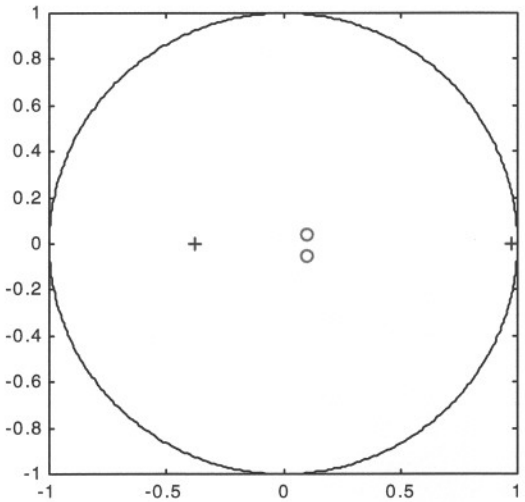
On a pu obtenir les résultats du régulateur RST Landau suivants :

- Fonction de transfert du régulateur RST Landau :

Transfer function:
-z^2 + 0.1906 z - 0.01137

0.2182 z^2 - 0.1308 z - 0.08

- Pôles et zéros du régulateur RST Landau:

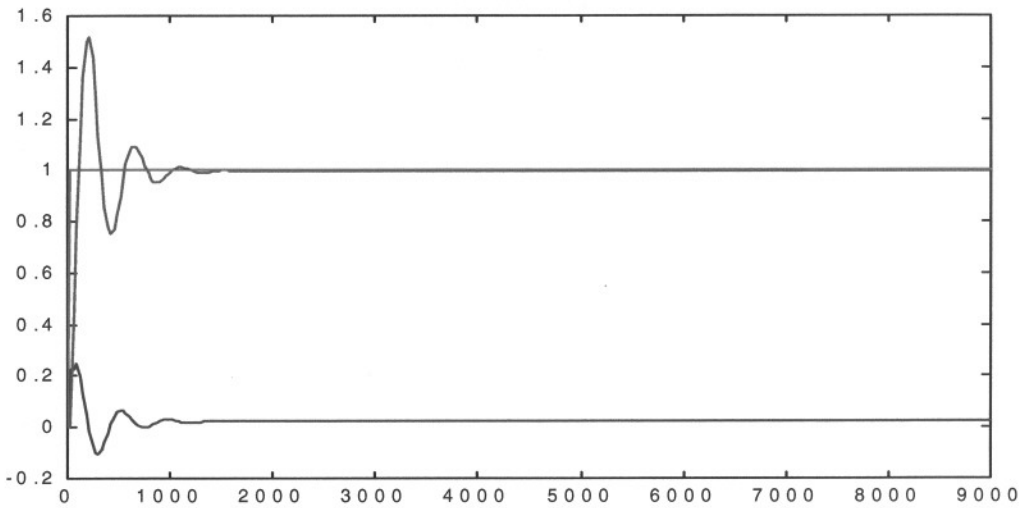


On peut observer deux zéros conjugués:
z1 = 9.5320e-002 +4.7761e-002i

z2 = 9.5320e-002 -4.7761e-002i
qui sont en accord avec la théorie du placement des pôles.
Et deux pôles :
P1 = 9.7545e-001

P2 = -3.7585e-001

Comportement dynamique du régulateur RST Landau :



3.4 Analyse des résultats obtenus des deux méthodes :

Rappelons tout d'abord les résultats des régulateurs RST calculés et le régulateur PID digital du CERN :

- Régulateur RST (méthode LoopShap CERN) :

  Transfer function:
  0.4083 z^3 + 0.01424 z^2 + 4.721e-005 z + 6.048e-008

  0.003669 z^4 + 0.03677 z^3 + 0.00078 z^2 + 3.209e-006 z

- Régulateur RST (méthode Landau) :

  Transfer function:
  -z^2 + 0.1906 z - 0.01137

  0.2182 z^2 - 0.1308 z - 0.08

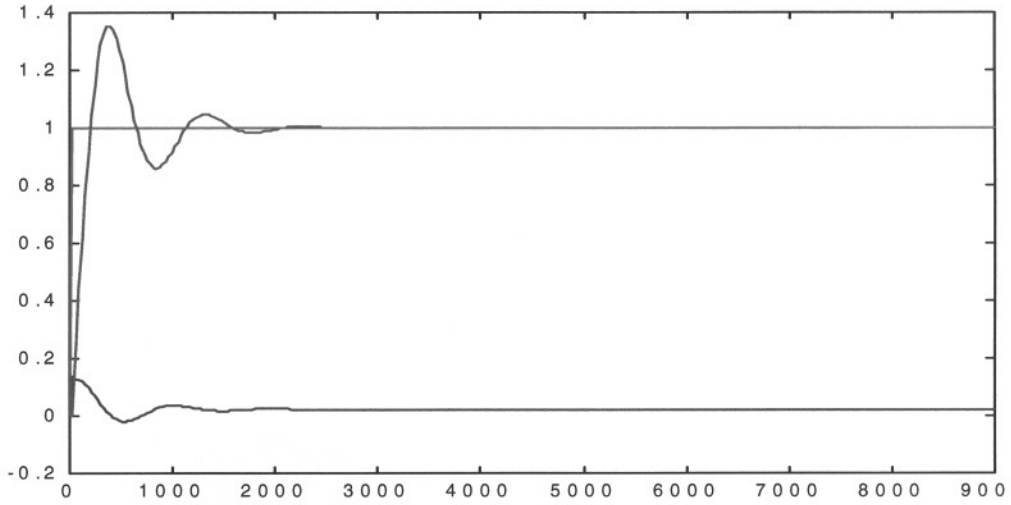
- Régulateur PID digital du CERN :

  Transfer function:

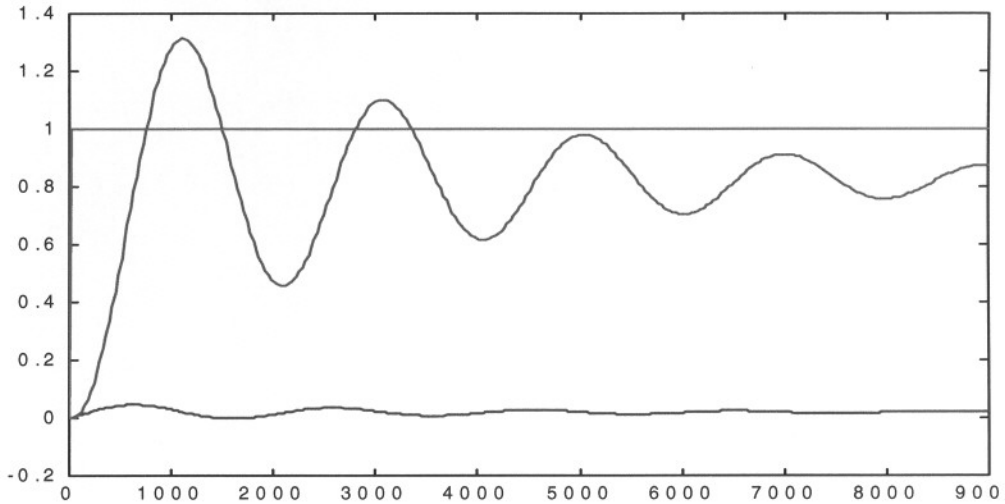
  -2.77 z^2 + 2.33 z – 0.17
  ---
  z^2 - z

On peut remarquer la différence d'ordre du régulateur RST Landau et celle du régulateur LoopShap CERN. La raison est la méthode Landau optimise les coefficients du régulateur RST calculé. En général, l'ordre du régulateur RST évalué par cette méthode ne dépasse pas l'ordre 2.
On peut maintenant comparer la réponse indicielle de chaque régulateur.

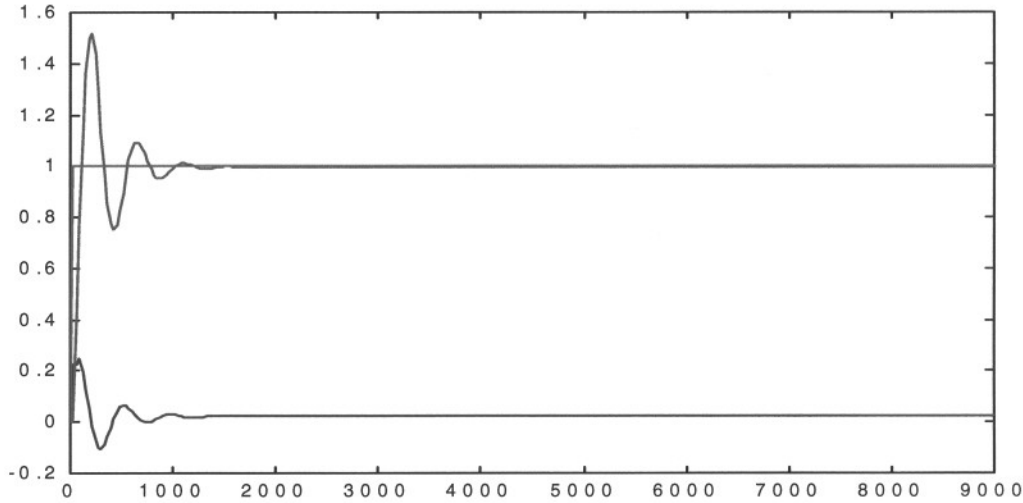
Réponse indicielle du régulateur PID digital du CERN



Réponse indicielle du régulateur RST LoopShap CERN



Réponse indicielle du régulateur RST Landau



On peut remarquer que les résultats sont concluants car la convergence vers la stabilité du régulateur RST est nettement plus rapide que celle du régulateur PID digital du CERN. Le temps de convergence pour le régulateur RST est d'environs de 16 minutes alors que pour le régulateur PID digital du CERN, il est de 41 minutes.
Ce temps de convergence vers la stabilité est très important pour un système de refroidissement comme l'échangeur de température du CERN. Au moment d'une perturbation extérieure ou une coupure du circuit, il est important que le régulateur puisse tout de suite réagir et adapter la régulation dont le système a besoin.

On dispose maintenant d'un régulateur RST (Landau en particulier) robuste que l'on va pouvoir implémenter sur un automate afin d'étudier sa réaction en simulation.

4. IMPLANTATION DU SYSTEME REGLE SUR L'AUTOMATE S7 :

4.1 Présentation du logiciel SIMATIC S7 V5.2 :

STEP 7 est le logiciel de base pour la programmation et la configuration dans SIMATIC. Il est formé d'un ensemble d'applications avec lesquelles vous pouvez aisément réaliser des tâches partielles comme :

- la configuration et le paramétrage du matériel (l'automate)
- la création et le test de programmes utilisateur

· la configuration de réseaux et de liaisons (TCP/IP, Ethernet, MPI, Profibus,..)

S'y ajoute une vaste gamme de logiciels optionnels, dont entre autres ceux des langages de programmation SCL, GRAPH ou HiGraph.

Le gestionnaire de projets SIMATIC, encore appelé SIMATIC Manager, sert d'interface graphique à toutes ces applications. C'est lui qui organise la mise en commun dans un projet de toutes les données et de tous les paramètres requis pour réaliser une tâche d'automatisation. Les données y sont structurées thématiquement et représentées sous forme d'objets.

Un projet STEP 7 est composé de dossiers et d'objets. On pourrait le comparer à une structure arborescente comportant des dossiers et des fichiers, comme celle connue dans l'Explorateur Windows. Un dossier désigne un objet qui lui-même peut comporter d'autres dossiers et objets, comme par exemple un programme S7 contenant les dossiers Blocs et Sources ainsi que l'objet "Mnémoniques".

En ce qui concerne les normes en vigueur, les langages de programmation SIMATIC intégrés à STEP 7 répondent à la norme DIN EN 6.1131-3.
Le progiciel de base s'exécute sous les systèmes d'exploitation MS Windows 95, MS Windows 98, MS Windows Millenium, MS Windows NT 4.0 Workstation, MS Windows 2000 Professional ainsi que MS Windows XP Professional et s'adapte à son organisation graphique orientée objet.

Ce logiciel de base vous assiste dans toutes les phases du processus de création de vos solutions d'automatisation, comme par exemple :

- la création et la gestion de projets,
- la configuration et le paramétrage du matériel et de la communication,
- la gestion des mnémoniques,
- la création de programmes, par exemple pour les systèmes cible S7,
- le chargement de programmes dans des systèmes cible,
- le test de l'installation d'automatisation,
- le diagnostic lors de perturbations de l'installation.

La conception de l'interface utilisateur du logiciel STEP 7 répond aux connaissances ergonomiques modernes et son apprentissage est très facile.

En ce qui concerne l'éditeur de mnémoniques, il vous permet de gérer toutes les variables globales. Vous disposez des fonctions suivantes :

- définition de désignations symboliques et de commentaires pour les signaux du processus (entrées/sorties), mémentos et blocs,
- fonctions de tri,
- importation/exportation avec d'autres programmes Windows.

La table des mnémoniques qui en résulte est mise à disposition de toutes les applications. La modification de l'un des paramètres d'un mnémonique est de ce fait reconnue automatiquement par toutes les applications.

Vous avez la possibilité d'avoir le diagnostic du matériel. Il fournit un aperçu de l'état du système d'automatisation. Dans une représentation d'ensemble, un symbole permet de préciser pour chaque module, s'il est défaillant ou pas. Un double clic sur le module défaillant permet d'afficher des informations détaillées sur le défaut. Les informations disponibles dépendent des différents modules :

- affichage d'informations générales sur le module (p.ex. numéro de commande, version, désignation) et son état (p.ex. défaillant),
- affichage d'erreurs sur les modules (p.ex. erreur de voie) de la périphérie centrale et des esclaves DP,
- affichage des messages de la mémoire tampon de diagnostic.

Pour les CPU, des informations supplémentaires s'affichent :

- causes de défaillance dans le déroulement d'un programme utilisateur
- durée de cycle (le plus long, le plus court et dernier),
- possibilités et charge de la communication MPI,
- performances (nombre d'entrées/sorties, de mémentos, de compteurs, de temporisations et de blocs possibles).

Les langages de programmation CONT, LIST et LOG pour S7-300/400 font partie intégrante du logiciel de base.

- Le schéma à contacts (CONT) est un langage de programmation graphique. La syntaxe des instructions fait penser aux schémas de circuits. CONT permet de suivre facilement le trajet du courant entre les barres d'alimentation en passant par les contacts, les éléments complexes et les bobines.
- La liste d'instructions (LIST) est un langage de programmation textuel proche de la machine. Dans un programme LIST, les différentes instructions correspondent, dans une large mesure, aux étapes par lesquelles la CPU traite le programme. Pour faciliter la programmation, LIST a été complété par quelques structures de langage évolué (comme, par exemple, des paramètres de blocs et accès structurés aux données).
- Le logigramme (LOG) est un langage de programmation graphique qui utilise les boîtes de l'algèbre de Boole pour représenter les opérations logiques. Les fonctions complexes, comme par exemple les fonctions mathématiques, peuvent être représentées directement combinées avec les boîtes logiques.

Vous pouvez vous procurer d'autres langages de programmation sous forme de logiciels optionnels.

En ce qui concerne la configuration matérielle, vous utilisez cette application pour configurer et paramétrer le matériel d'un projet d'automatisation. Vous disposez des fonctions suivantes :

- Pour configurer le système d'automatisation, vous sélectionnez des châssis (Racks) dans un catalogue électronique et affectez les modules sélectionnés aux emplacements souhaités dans les racks.
- La configuration de la périphérie décentralisée est identique à celle de la périphérie centralisée. La périphérie voie par voie est également possible.
- Pour le paramétrage de la CPU, des menus vous permettent de définir des caractéristiques telles que le comportement à la mise en route et la surveillance du temps de cycle. Le fonctionnement multiprocesseur est possible. Les données saisies sont enregistrées dans des blocs de données système.
- Pour le paramétrage des modules, des boîtes de dialogue vous permettent de définir tous les paramètres modifiables. Les réglages à l'aide de commutateurs DIP s'avèrent inutiles. Le paramétrage des modules est réalisé automatiquement au démarrage de la CPU. L'avantage suivant en résulte. Le remplacement d'un module est ainsi possible sans nouveau paramétrage.
- Le paramétrage de modules fonctionnels (FM) et de processeurs de communication (CP) s'effectue de manière identique à celui des autres modules dans la configuration matérielle. A cet effet, des boîtes de dialogues ainsi que des règles spécifiques aux modules sont ainsi mises à disposition pour chaque FM et CP (fournies dans le logiciel fonctionnel du FM/CP). Dans les boîtes de dialogue, le système ne propose que des saisies possibles, ce qui empêche les entrées erronées.

4.2 Programmation sur SIMATIC S7 V5.2 du système régler :

La programmation s'est effectuée essentiellement en langage LIST et SCL.
On dispose de deux programmes :

- Le premier programme OB1 en langage LIST:

"Periode d'echantillonnage Te=30s"

```
UN   M      0.1
L    S5T#30S
SE   T      1
U    T      1
=    M      0.1
U    M      0.1
CC   FC     1
```

Il s'agit d'un ''Timer'' permettant de définir une période d'échantillonnage Te de 30 [s] pour la simulation. Il décompte de 30 à 0 et arrivé à 0, la condition CC permet d'activer la fonction FC 1.

- Le deuxième programme ''RegRST.scl'' : (voir annexe 3)
  Il possède 4 blocs de données DBxx qui contiennent les variables globales du programme (Paramètres RST, entrées et sorties du régulateur RST, paramètres du processus) et la fonction FC1 (intitulé dans le premier programme. L'algorithme utilisé pour la régulation RST est identique à celle définit au chapitre Définition du régulateur RST (cf chapitre 3.1).
  Il ne s'agit encore ici que d'une ébauche du programme de régulation. Pour information, il n'existe pas encore de module RST préprogrammé dans la bibliothèque de Siemens.
  La question à se poser serait, pourquoi avoir implémenter le régulateur RST et le processus dans l'automate. La raison est qu'il faut disposer du logiciel OGP qui permet de visualiser sur l'interface utilisateur le comportement dynamique du régulateur et du processus. Ce n'est pas le cas.

Cette partie termine les différentes étapes énoncées du projet.
La simulation sur l'automate sera présentée lors de la présentation du projet de diplôme.

5. CONCLUSION :

Le cahier des charges a été en grande partie respecté. L'identification du processus de l'échangeur de température a permis de comprendre son fonctionnement et les paramètres qui sont mis en jeu. La notion d'identification de processus a paru au début du projet pas très claire. Mais, lorsque que l'on dispose d'informations complètes du dispositif réel, le déroulement de l'identification suit son cours.
Il n'a pas été évident d'obtenir les mesures adéquates à temps au début du projet, car les tâches administratives pour mon inscription à l'enceinte du CERN ont pris beaucoup de temps, ainsi que le matériel nécessaire mis à ma disposition. Il est important de posséder une carte d'accès pour entrer dans les bâtiments auxiliaires BA's pour la prise des mesures. L'identification du processus a été insignifiante au départ mais elle a permis de remettre en cause la compréhension du processus.
Après avoir établi une bonne identification, le dimensionnement d'un régulateur RST pour la régulation de ce processus s'en est suivi. On a pu obtenir de très bons résultats avec deux méthodes différentes. Il est important d'appliquer au moins deux stratégies de dimensionnement car elles permettent de vérifier nos résultats et d'assurer que l'on ne se dirige pas sur une mauvaise voie. Il faut cependant consacrer beaucoup de

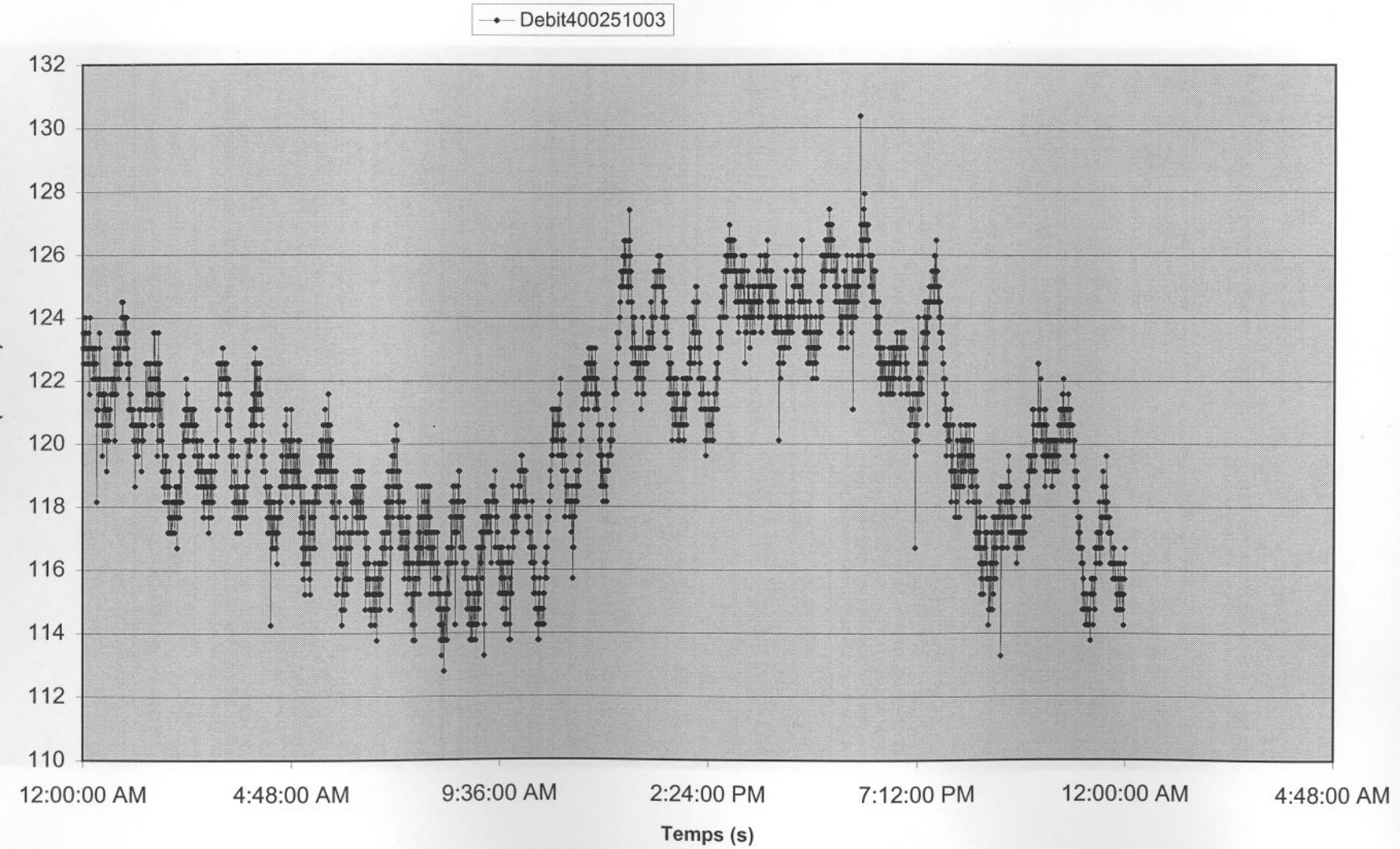
temps à ces deux étapes (identification et dimensionnement) car il serait inutile d'implémenter un régulateur RST et processus qui n'ont aucun lien avec la réalité. L'implémentation sur l'automate reste la dernière étape du projet mais le délai du travail de diplôme est à terme. C'est pour cette raison, que la simulation sera présentée lors de l'exposé du projet pour l'obtention du diplôme.

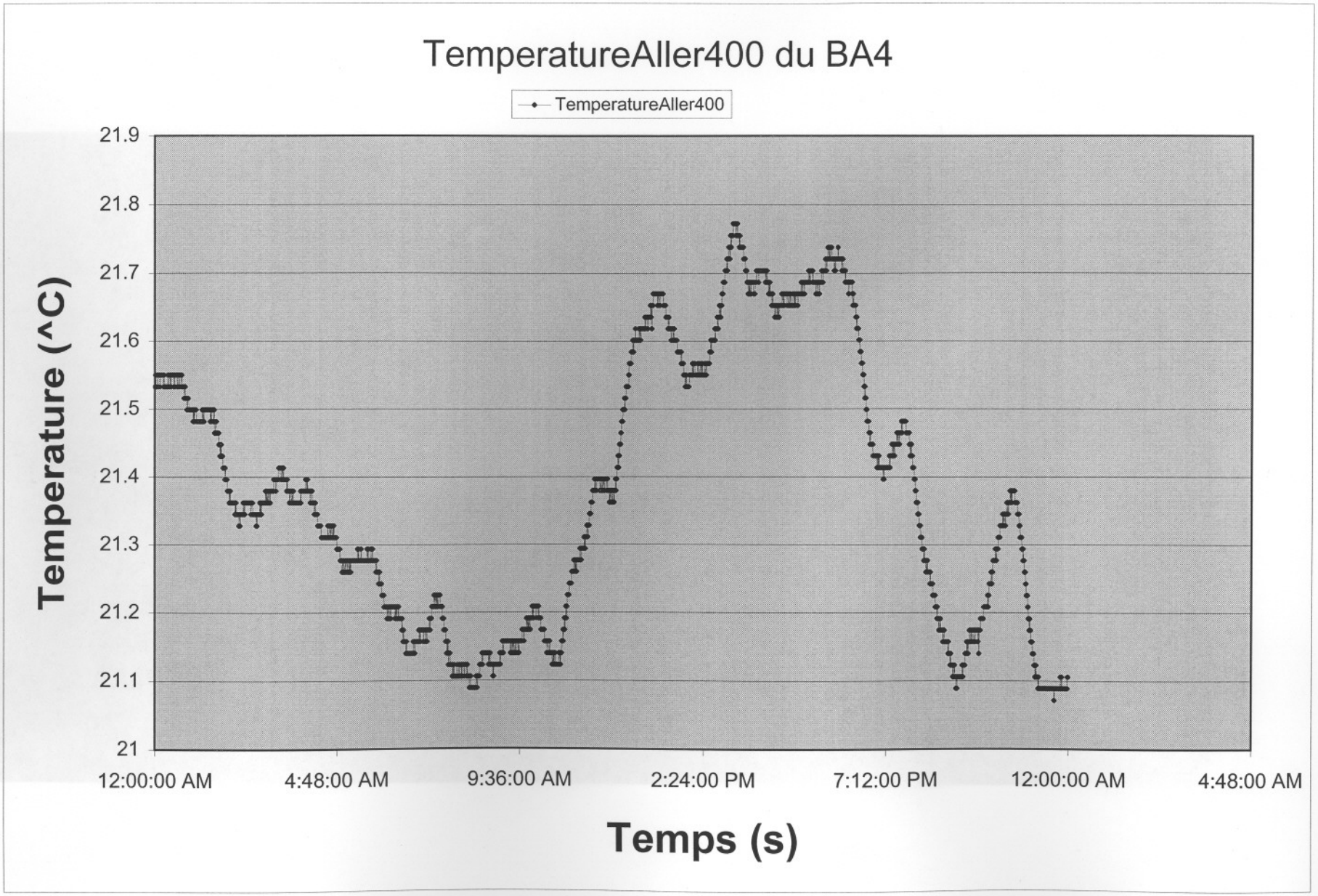
Ce projet de diplôme a permis d'apprendre le terme de conscience professionnelle et les responsabilités auxquelles il est rattaché. Dans une grande entreprise, comme le CERN, nous sommes livrés à nous même et donc à nous de prendre les initiatives pour l'élaboration du travail de diplôme. Il faut communiquer avec les personnes concernées, effectuer des recherches dans la littérature, se remettre en cause lors d'une impasse. Ce projet sera pour la section ST/CV une avancée pour l'évolution des échangeurs de température en intégrant dans les temps à venir le régulateur RST dans les automates S7. Ce serait une tâche très intéressante pour l'intégration dans la vie professionnelle pour ma part.

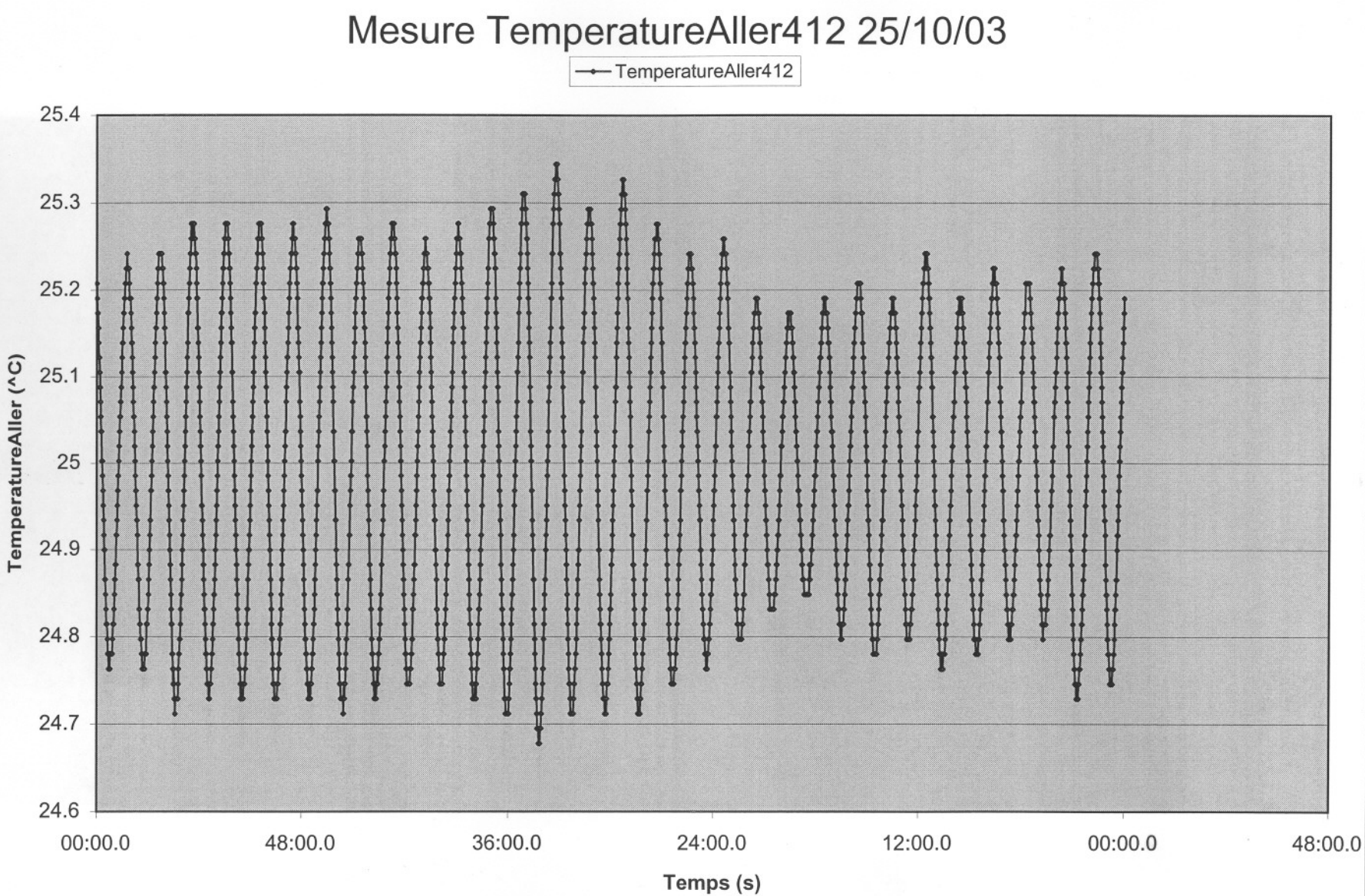
ANNEXE N°1 :

- **graphique Excel : Débit 400 du circuit primaire**
- **graphique Excel : Température Aller 400 circuit primaire**
- **graphique Excel : Température Aller 412 circuit secondaire**

Mesure du debit BA4 (circuit primaire) 25/10/03




TemperatureAller400 du BA4
TemperatureAller400
21.9
21.8
21.7
21.6
21.5
21.4
21.3
21.2
21.1
21
Temperature (^C)
12:00:00 AM
4:48:00 AM
9:36:00 AM
2:24:00 PM
7:12:00 PM
12:00:00 AM
4:48:00 AM
Temps (s)


Mesure TemperatureAller412 25/10/03
TemperatureAller412
25.4
25.3
25.2
25.1
25
24.9
24.8
24.7
24.6
TemperatureAller (^C)
00:00.0
48:00.0
36:00.0
24:00.0
12:00.0
00:00.0
48:00.0
Temps (s)

ANNEXE N°2 :

- programme Matlab : LoopShapCERN
- programme Matlab : ParamCERN.m
- programme Matlab : ouvdiscret.m
- programme Matlab : cartediscret.m

```
%Exemple CERN      Façonnage de Boucle
%
%performance
omegarM=0.1/4;rM=0.66;[numw1,denw1]=zp2tf(-4/omegarM,[0 0],rM*omegarM);
[numw11,denw11]=zp2tf([-omegarM*rM/4;-omegarM/8;-0.51],[0;0;-0.70],1);
omega1=logspace(log10(omegarM/40),log10(omegarM/4),40);
%omega11=logspace(log10(omegarM/100),log10(omegarM/0.01),40);
%figure(3);freqs1(numw11,denw11,omega11);hold on;

%incertitude
[numw2,denw2]=zp2tf(-1,[0 -1 -10],1e2);
[numw22,denw22]=zp2tf([-0.02;-1],[-1e-2;-1e1],1);
omega2=logspace(0,1,20);
%omega22=logspace(-3,1,20);
%figure(3);freqs1(numw22,denw22,omega22);%hold off;

%Limite inférieure aux basses fréquences
denw12=denw22-numw22;denw12=conv(denw12,denw11);
numw12=conv(denw22,numw11-denw11);
figure(1);[a,b]=freqs1(numw12,denw12,omega1);
loglog(b,abs(a));axis([1e-3 1e1 1e-2 1e2]);hold on

%Limite supérieure aux hautes fréquences
numw21=denw11-numw11;numw21=conv(numw21,denw22);
denw21=conv(denw11,numw22);
figure(1);[c,d]=freqs1(numw21,denw21,omega2);
loglog(d,abs(c));

%système à régler
dens=[1.0000e+000  3.4876e-002  1.1563e-004  1.4814e-007];
nums=[3.6688e-003  7.7969e-005  3.2089e-007];%

%système en boucle fermée
[num0,den0]=zp2tf([],[0;-1/0.10],0.40825);%
omega=logspace(-3,1,100);[e,f]=freqs1(num0,den0,omega);loglog(f,abs(e));hold off;
set(gca,'XTick',[0.001 0.01 0.1 1 10],'YTick',[0.01 0.1 1 10 100]);grid on;
line('LineStyle', ':', 'Color', 'r', 'XData',[1e-2,1e0], 'YData', [1e1,1e-1]);%
figure(2);semilogx(f,180*phase(e)/pi);
%calcul régulateur
numR=conv(num0,dens)%
denR=conv(den0,nums)%
figure(3);pzmap(numR,denR);graph_0;
[zeroR,poleR,gainR]=tf2zp(reduirevect(numR),reduirevect(denR));
```

```
%Paramètres de régulateur CERN discret

%PID actuel
pidS =[-2.7700e+000  2.3300e+000 -1.7000e-001];
pidR =[1    -1     0];
%RST-Landau
lanR =[-2.1821e-001  1.3084e-001  8.0001e-002];
lanS =[1.0000e+000 -1.9064e-001  1.1367e-002];

%RST-LoopShap CERN
loopR =[0.003669 0.03677  0.00078 3.209e-006 0];
loopS =[0.4083 0.01424 4.721e-005 6.048e-008];
```

```
% Conversion de boucle ouverte en discret
[num0z,den0z]=c2dm(num0,den0,TE,'zoh');
num0z=num0z;%/4;
figure(6);dbode(num0z,den0z,TE);
numsz=[0.09348,-0.1381,0.0499];
densz=[1,-2.286,1.64,-0.3512];
numRz=-conv(num0z,densz);denRz=conv(numsz,den0z);
figure(6);dbode(num0z,den0z,TE);
```

```
[pdiscret,zdiscret]=pzmap(numRz,denRz);
figure(5);
plot(real(pdiscret),imag(pdiscret),'+b',real(zdiscret),imag(zdiscret),'or');
cerclunite(5)
```

ANNEXE N°3 :

Programme Simatic V7 5.2 : RegRST.scl

gulateur RST'

de l'echangeur de temperature (BA4)

.0'
AIRE Sebastien

internes du regulateur RST

```
10 DATA_BLOCK    DB01
11 STRUCT  //  Description des polynomes R, S et T constituant le regulateur RST
12     R : ARRAY   [0..2] OF  REAL;
13     S : ARRAY   [0..2] OF  REAL;
14     T : ARRAY   [0..2] OF  REAL;
15 END_STRUCT
16 BEGIN
17 // Affectation des coefficients du polynome R
18 R[0]:=8.0001E-2  ;
19 R[1]:=1.3084E-1  ;
20 R[2]:=-2.1821E-1  ;
21 // Affectation des coefficients du polynome S
22 S[0]:= 1.1367E-2 ;
23 S[1]:= -1.9064E-1 ;
24 S[2]:= 1.0;
25 // Affectation des coefficients du polynome T
26 T[0]:= 1.0 ;
27 T[1]:= 0.0;
28 T[2]:= 0.0;
29
30 END_DATA_BLOCK
31
32 // Parametres d'entree  du regulateur RST
```

```
33 DATA_BLOCK    DB02
34 STRUCT //  Description des variables d'entree regulateur RST
35    Ymes  : REAL;
36    Cons  : REAL;
37 END_STRUCT
38 BEGIN
39 // Affectation de la sortie Ymes
40 Ymes:=25.31;
41 // Affectation de la consigne Cons
42 Cons:=25.0;
43 END_DATA_BLOCK
44
45 // Parametres de sortie du regulateur RST
```

```
46 DATA_BLOCK    DB03
47 STRUCT  //  Description de la variable de sortie du regulateur RST
48 Ucm  : REAL ;
49 END_STRUCT
50 BEGIN
51 // Affectation de l'organe de commande Ucm
52 Ucm:=0.0;
53 END_DATA_BLOCK
54
55
```

```
56 DATA_BLOCK    DB04
57   STRUCT  //  Description du numerateur B et denominateur A du processus du systeme
58   NumB  : ARRAY   [0..2] OF REAL ;
59   DenA  : ARRAY   [0..2] OF REAL ;
60 END_STRUCT
61 BEGIN
62 // Affectation des coefficients du numerateur NumB du processus
63 NumB[ 0]:= -7.8676E-8 ;
64 NumB[ 1]:= 5.8578E-6 ;
65 NumB[ 2]:= 0.0;
66 // Affectation des coefficients du denominateur DenA du processus
67 DenA[ 0]:= 5.1705E-6 ;
68 DenA[ 1]:= 3.1557E-4 ;
69 DenA[ 2]:= 1.0;
70
71 END_DATA_BLOCK
72
73
74
75
76
```

```
77 FUNCTION   FC1 : VOID
78 VAR_TEMP
79   // réservé
80   info : ARRAY [0..19] OF BYTE;
81   // Variables temporaires
82   //Variables temporaires
83   // Commande du regulateur pour la mise a jour
84   u:REAL;
85   // Mesure de sortie du processus de l'echangeur pour la mise a jour
86   y:REAL;
87   // Variables pour le calcul de la mise a jour de la commande.
88   sigmar: REAL;
89   sigmas: REAL;
90   sigmat: REAL;
91   i:INT ;
92   // Variables pour le calcul de la mise a jour de la mesure de sortie
93   sigmab: REAL;
94   sigmaa: REAL;
95
96 END_VAR
97 // Programme principal
98 BEGIN
99
100
101 i:=0;
102 // Affectation des variables d'entree du regulateur.
103 y:=DB02.Ymes;
104 u:=DB03.Ucm;
105 //
106 // Partie regulateur RST
107 //
108 // Calcul de la commande a appliquer
109 sigmar :=DB01.R[0]*y;
110 sigmas :=0;
111 sigmat :=DB01.T[0]*DB02.Cons;
112
113 // Boucle de calcul
114 FOR  i:=1 TO  2 BY  1 DO
115     sigmar :=sigmar  + DB01.R[i] *y;
116     sigmas :=sigmas  + DB01.S[i] *u;
117 END_FOR  ;
118 // Calcul de la commande actuelle pour la sortie du regulateur
119     u :=(sigmat  - sigmar  - sigmas) /(DB01.S[2]);
120
121     DB03.Ucm:=u;
122 //
123 // Partie du processus de l'echangeur de temperature
124 //
125  sigmab  := DB04.NumB[0]  * DB03.Ucm;
126  sigmaa  := DB04.DenA[0]  * DB03.Ucm;
127   // Boucle de calcul
128 FOR  i:=1 TO  3 BY  1 DO
129     sigmab :=sigmab  + DB04.NumB[i] *DB03.Ucm;
130     sigmaa :=sigmaa  + DB04.DenA[i] *DB03.Ucm;
131 END_FOR  ;
132
133 // Calcul de la commande actuelle pour la sortie du regulateur
134 DB02.Ymes :=sigmab /sigmaa;
135
136
137 END_FUNCTION
138
139
```

BIBLIOGRAPHIE :

- ''Identification des processus''. L.Povy - édition Dunod
- ''System Identification Toolbox Matlab''. Lennart Ljung – édition M.W
- ''Process Control Matlab''. Pao C. Chau – édition Cambridge
- ''Commande automatique des systèmes linéaires continus''. Viorel Minzu et Bernard Lang – édition ellipse
- Divers classeurs de Adaptech sur le régulateur RST, la robustesse, l'identification de processus. I.D Landau